International Journal of Applied Science
ISSN (Online): (ISSN: 2208-2182)
Volume 10 Issue 01-March, 2024
DOI: 10.53555/eijas.v10i1.190

REVIEW OF MATLAB-BASED PROJECTS FOR UNDERSTANDING HALF-WAVE DIPOLE ANTENNA CHARACTERISTICS

Sonu Rani*

**PHd physics, Om Sterling Global University, Hissar (Haryana)*

Corresponding Author:

Abstract

This article gives a detailed analysis of how MATLAB is used to model half-wave dipole antennas, a basic component in antenna theory and wireless communication. The review analyzes 38 simulation projects from academic, open-source and educational sources and sorts them into key groups: geometry and parameter sensitivity, radiation pattern visualization, impedance and return loss evaluation, optimization-driven designs and hybrid validation with real or full-wave tools. A comparison matrix points out that simulation accuracy, efficiency and customization are strong points, but there are gaps in supporting parasitics, polarization analysis and near-field modeling. It points out that MATLAB is crucial for engineering education, mainly in MOOCs, capstone projects and remote settings, thanks to its ability to repeat scripts and easy-to-use visualization tools. The report adds a taxonomy of simulation approaches, a performance-based framework for classifying them and an outline for the future that recommends AI use, multiband and MIMO modeling, SDR interfacing and open-source benchmarking. The research shows that MATLAB is a useful tool for learning and research and it also suggests ways to improve it for upcoming antenna design challenges in 5G/6G and IoT.

Keywords: *MATLAB simulation, dipole antenna, antenna education, impedance matching, optimization techniques.*

2. Introduction

Radio frequency systems rely heavily on half-wave dipole antennas which are now essential in designing modern communication networks. Because they are straightforward, have good resonance and a dependable pattern of radiation, these antennas are widely studied and used in wireless communication, radar and IoT (Balanis, 2016; Paine, 2017). Since 5G and 6G networks are using more D2D technology and IoT is expanding, there is now a strong demand for reliable, efficient and compact antennas that rely on classical theory (Gismalla et al., 2022; Jahanbakhsh Basherlou et al., 2025).

With the progress in technology, MATLAB has come to be important for examining and visualizing how antennas function. The MATLAB Antenna Toolbox which MathWorks has been improving over time, allows for effective modeling of antennas like dipoles, monopoles, arrays and reflectors, covering both basic and advanced tasks (Makarov et al., 2021). Because MATLAB supports analytical modeling, FEA and algorithmic design, researchers and teachers can easily simulate different electromagnetic situations. It is possible for users to define geometric details, get return loss (S11), impedance, gain and view both 2D and 3D radiation patterns using simple commands. It is also useful that MATLAB supports adding custom code or machine learning algorithms to its simulation tools which is helpful for both academic and practical antenna research (Sarker et al., 2023; Rashid & Singh, 2024).

Although MATLAB plays a big role in teaching and research on antennas, there is not yet a comprehensive review that focuses on using MATLAB for project-based modeling of half-wave dipole antennas. Many existing studies are found in theses, technical blogs, code repositories and single case studies and they are not well-organized or compared (Ramesh et al., 2020). Usually such isolated studies fail to compare the different methods used in modeling, simulation, visualization or optimization. Also, these projects are rarely assessed in terms of how well they help move from theory to practice, especially in schools or when developing prototypes.

The main reason for this review is the gap that was found. Since half-wave dipole antennas are widely applied for both study and research, it is important to review how MATLAB is being employed for technical modeling, teaching and practical testing. Because HFSS and CST Microwave Studio are now available as alternative high-fidelity solvers, it's important to compare MATLAB with these tools in terms of flexibility, reproducibility and accessibility (Elwasife et al., 2023; Chen et al., 2024).

So, this review article begins with two important goals in mind:

1. To synthesize and evaluate project-based dipole simulations developed in MATLAB, focusing on their modeling techniques, parameterization strategies, radiation pattern analysis, impedance matching procedures, and algorithmic optimization mechanisms.
2. To bridge the gap between theoretical understanding and practical implementation by critically examining how these projects support intuitive learning, conceptual visualization, and simulation fidelity in both academic and applied engineering contexts.

They will be achieved by studying MATLAB-based projects found in academic journals, conference proceedings, open-source repositories and instructional materials. The review looks at projects that work with MATLAB's Antenna Toolbox and at those that feature custom electromagnetic solvers or optimization routines. Emphasis is on evaluating various modeling methods, the results of simulations such as return loss and radiation patterns and the use of deep learning for parameter adjustment and better performance (Sarker et al., 2023; Rashid & Singh, 2024).

The review also looks at how MATLAB supports learning in schools, since interactive and visual simulations are now a main way to explain antenna concepts. With MATLAB, students can change geometric parameters, check real-time results and see the link between theories and real performance which is often hard to do with just analytical approaches (Makarov et al., 2021; Ramesh et al., 2020).

3. Theoretical Foundations of Half-Wave Dipole Antennas

The half-wave dipole antenna is a basic part of radiating systems and has been important in advancing modern antenna technologies. It acts as a guide for studying complex geometries and is popular in wireless communication, sensing technology, RFID and biomedical fields because of its simplicity, good resonance and defined properties (Anam et al., 2025; Wang et al., 2024).

3.1 Radiation Physics and Current Distribution

A half-wave dipole antenna uses two conductive elements that are fed in the center and it works at a frequency where the total length is about half the wavelength of the signal. Alternating current causes the current in each arm of the dipole to increase at the feed point and decrease to zero at the ends in a sinusoidal pattern. The uneven distribution of current causes the radiation to have a toroidal (donut-shaped) pattern, with the strongest field directed broadside to the axis and no radiation at the ends (Hirani et al., 2019). Figure 1 shows how the current pattern on a toroidal dipole and its radiation pattern are related.

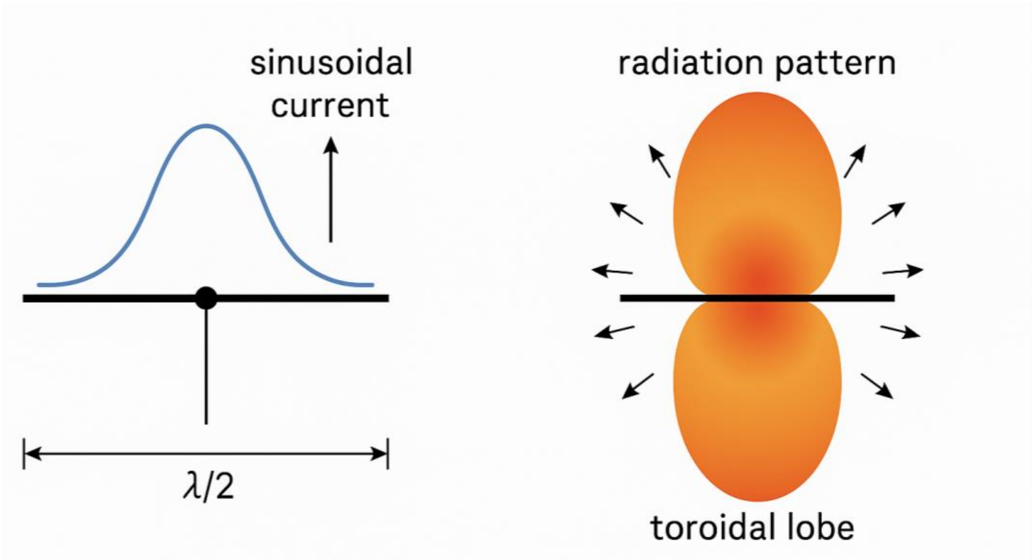

Figure 1. Current and radiation characteristics of a half-wave dipole antenna under ideal conditions.

3.2 Resonant Length Derivation ($L \approx \lambda/2$)

The dipole antenna resonates when its length is about half the wavelength (λ/2) of the signal in the medium it is in. If everything is ideal, the antenna produces a standing wave where the highest current appears at the center. Still, because of end effects and reactive loading, the true electrical length is usually a bit less than λ/2, falling in the range of 0.48λ to 0.49λ (Lo & Lee, 2013). In free space, the basic design equation for a resonant dipole is:

$$L = \frac{c}{2f\sqrt{\varepsilon_{eff}}}$$

where c is the speed of light, f is the resonant frequency, and ε_{eff} is the effective relative permittivity of the surrounding medium. In practice, accurate determination of L often requires simulation tools or empirical adjustments, especially in embedded or wearable environments (Wang et al., 2024).

3.3 Key Performance Parameters

Many factors determine how well a half-wave dipole antenna works. Return loss, bandwidth, gain, directivity and radiation efficiency are all important factors that affect how useful an antenna is in real life (see Table 1). By correctly modeling and simulating these parameters, you can fine-tune and improve performance, mainly in MATLAB-based workflows:

- Return Loss (S11): This is the result of how much power is lost because the antenna and feed do not match well. A center-fed dipole at resonance usually has return loss values lower than -15 dB, suggesting very little reflection. It is still very important to have accurate impedance matching in real situations, especially when dealing with compact or multi-material platforms (Monica, 2020).
- Bandwidth: Half-wave dipoles have narrow bandwidths (around 3-5% of the central frequency), but this can be increased by making the dipole arms thicker or adding parasitic elements. Some recent designs make use of fractal shapes and wideband feeds to make antennas better without making them bigger (Mathew & Ponnapalli, 2023).
- Gain and Directivity: The gain of an ordinary λ/2 dipole in free space is about 2.15 dBi which includes both directivity and efficiency. Directivity means the antenna focuses its energy in a particular direction and gain accounts for any losses and problems with matching. Changing the dipole design in base stations or 5G arrays can significantly increase gain (Zong et al., 2017; Liu et al., 2021).
- Radiation Efficiency: It shows the amount of power sent out compared to the power in, usually close to 100% for wire dipoles when everything is ideal. But, in real use, efficiency can be reduced by conductor losses, mismatched impedance or the absorption of signals by the environment, mainly in materials-integrated or wearable devices (Anam et al., 2025).

Table 1: Key Performance Parameters of Half-Wave Dipole Antennas

Parameter	Description	Typical Range / Value	Simulation Relevance
Return Loss (S11)	Power reflected back due to impedance mismatch	Below –15 dB at resonance	Crucial for validating feed-point matching and ensuring efficient energy transmission [(Monica, 2020)]
Bandwidth	Range of frequencies over which antenna performs acceptably	~3–5% of center frequency	Influences frequency tolerance; enhanced via design tweaks or fractal/parasite loading [(Mathew & Ponnapalli, 2023)]
Gain	Ratio of radiated power in the strongest direction to input power	~2.15 dBi (typical half-wave dipole)	Helps evaluate how well energy is focused; vital in directional arrays or MIMO designs [(Zong et al., 2017)]
Directivity	Ability to radiate in a specific direction relative to an isotropic source	~2.1 dB for half-wave dipole	Determines beam shaping and target coverage in pattern simulation [(Liu et al., 2021)]
Radiation Efficiency	Ratio of radiated power to total input power, accounting for losses	~85–100% (ideal dipole in free space)	Decreases with lossy materials or poor matching; affects energy budget in wearable systems [(Anam et al., 2025)]

3.4 Challenges in Real-World Realization

Even though half-wave dipoles have a simple theory, making them in real life is not easy:

- **End Effects:** Because the fringing fields at the open ends of the dipole affect the electrical length, it is necessary to correct or tune the design. This is most noticeable in miniaturized and chipless RFID antennas, where the shape of the antenna matters the most (Anam et al., 2025).
- **Feed-Point Impedance Variability:** The feed-point impedance of the ideal center-fed half-wave dipole is about 73 ohms. But, this value depends a lot on the ratio of diameter to length, nearby objects, dielectric loading and how the antenna is fed. Off-center feeding and impedance matching networks are used to reduce mismatch losses (Liu et al., 2021).
- **Environmental Coupling and Detuning:** When using antennas in wearable and nanomaterial systems, they might couple with human tissue or other surfaces which results in changes in their frequency, radiation and overall efficiency. Because of these coupling effects, it is necessary to use advanced materials and bio-compatible substrates in modeling (Wang et al., 2024).
- **Complex Optimization Requirements:** The next-generation communication standards (5G/6G) require antennas to be resonant, as well as small, adjustable and capable of working on various bands. As a result, researchers have been exploring ways to use machine learning to adjust the shape of the dipoles in order to meet several key objectives (Pandey & Singh, 2024).

All in all, the theoretical features of half-wave dipole antennas give a good basis for exploring them through simulations. It is important to know these parameters when checking the accuracy of return loss, radiation patterns and gain in MATLAB-based modeling. It then explains how MATLAB assists in theoretical modeling by providing specialized toolboxes and script-based tools.

4. MATLAB as a Computational Platform for Antenna Simulation

MATLAB has become the main tool for antenna modeling, analysis and visualization. In the area of half-wave dipole antennas, MATLAB allows engineers and educators to use automatic tools as well as program their own code to create, simulate and test various radiating structures. It has graphical interfaces that are user-friendly, along with low-level programming which means it can do everything from basic dipole modeling to advanced AI-based array development and quick deployment on hardware.

4.1 Antenna Toolbox Capabilities

Users can use MATLAB's Antenna Toolbox to model various antennas, for example, dipoles, monopoles and patch antennas, using object-based commands. Dipole antennas can be quickly prototyped and analyzed in real time using the dipole, impedance, returnLoss and pattern functions over frequencies set by the user. The toolbox helps you see radiation patterns in both 2D and 3D, calculate impedance and VSWR and export geometry for PCB layout use (Beswick, 2020).

Besides, the toolbox enables users to simulate antenna arrays and perform parametric sweeps to study how changes in geometry affect the antenna's resonant frequency, gain and direction. In schools and universities, this environment gives students a simple way to get accurate results, even though it is not as powerful as full-wave solvers. In addition, MATLAB performs frequency-domain analysis using the Method of Moments (MoM) which meets the accuracy requirements for wire antennas such as the half-wave dipole (Rahmat-Samii & Werner, 2025).

Figure 2 illustrates that users have the ability to set up a dipole's geometry, choose its feed spot and instantly see the radiation pattern at chosen frequencies using the GUI.

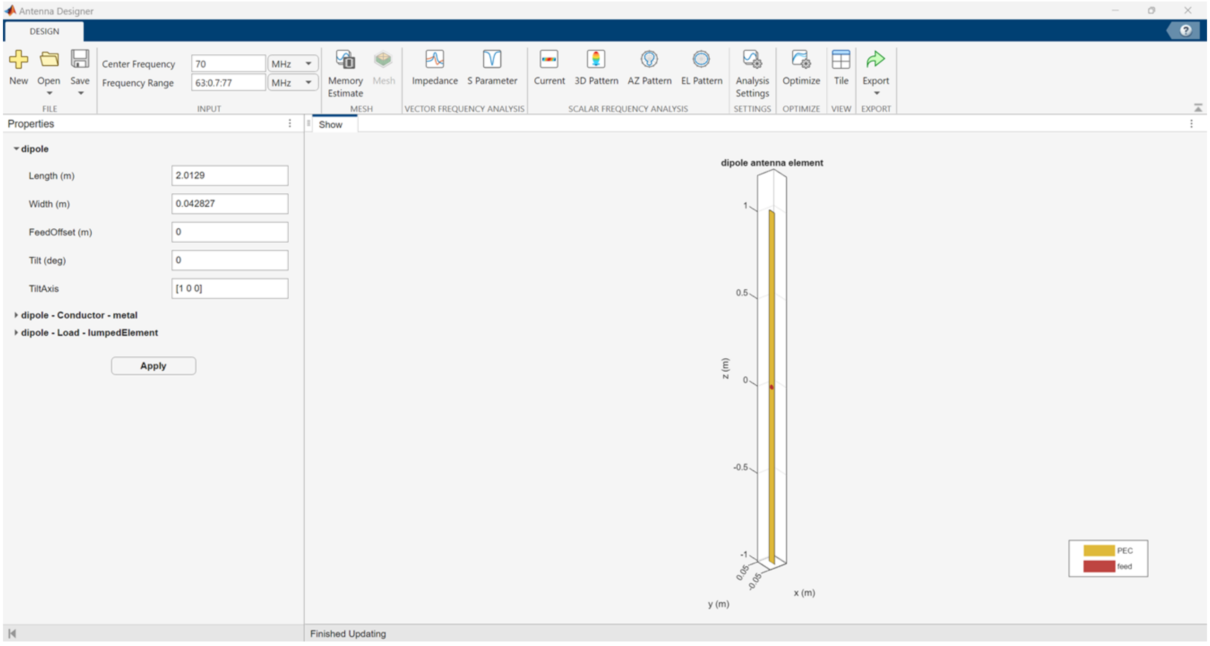

Figure 2. MATLAB Antenna Designer GUI showing dipole configuration and radiation pattern outputs. (Antenna Designer - Design, Visualize, and Analyze Antennas - MATLAB, n.d.)

4.2 Custom Script-Based Modeling

For those who want to use advanced electromagnetism methods or create their own solvers, MATLAB offers complete support for methods such as Finite Difference Time Domain (FDTD) and MoM. You can program these methods from the beginning using matrix operations and loops which makes it possible to run Maxwell's equations in MATLAB. This technique offers a clear and useful insight into how fields act on objects, how radiation works and the effects of boundaries which is of great help in engineering education (Notaroš et al., 2021).

Custom simulations make it possible to study antennas with new shapes, like those made from metamaterials or layered structures. Recently, using FDTD models in MATLAB has made it possible to simulate solar and wearable electromagnetic absorbers (Du John et al., 2024). Before going to commercial solvers or fabrication, these custom models are used to test design theories.

4.3 MATLAB vs. Full-Wave Solvers: Pros, Cons, and Use Cases

Comparing MATLAB to commercial solvers like CST and HFSS, it is clear that each program has its own advantages and disadvantages related to the project, the user and the system's complexity (see Table 2). With MATLAB, you can use transparent and flexible scripts and CST and HFSS are best for accurate modeling for industrial use. For this reason, full-wave solvers play a vital role in the final stages of testing industrial-grade antennas (Shamshad & Amin, 2012).

Table 2: Comparative Summary of MATLAB vs. Full-Wave Solvers

Tool	Accuracy	Runtime Efficiency	GUI Support	Custom Scripting	Hardware Needs
MATLAB	Moderate–High for wire antennas (MoM, FDTD custom codes)	High (lightweight, ideal for prototyping)	Antenna Designer App, Live Scripts	Excellent (supports algorithmic design)	Low–Moderate (runs on most PCs/laptops)
CST	Very High (adaptive meshing, full-wave solver)	Low (resource-intensive simulations)	Advanced GUI with 3D modeling and meshing	Limited (GUI-dominant with few scripting APIs)	High (requires GPU/CPU acceleration)
HFSS	Very High (finite element method with multiphysics)	Low–Moderate (depends on geometry)	Intuitive GUI with material customization	Limited scripting (uses Python/VBA)	High (optimized for high-performance setups)

Even so, these solvers are often hard to learn and expensive to use. Instead, MATLAB is especially good at prototyping, testing algorithms and validating concepts, allowing for much faster iteration. Because of its scripting environment, it can be used for doing multiple simulations, optimization and integrating different systems. In addition, MATLAB shows all the intermediate calculations which are usually hidden in full-wave solvers. This makes the process very helpful in research using optimization or real-time co-simulation (Musa et al., 2024).

Also, with MATLAB, it is possible to optimize antenna parameters by including algorithms such as genetic algorithms, differential evolution and particle swarm optimization, either natively or with the help of toolboxes, to achieve objectives such as improving gain or bandwidth (Rahmat-Samii & Werner, 2025).

4.4 Extensibility: GUI, Live Scripts, Simulink, and Code Generation

MATLAB is especially valued for its ability to be extended in many ways. Since the Antenna Designer App uses a GUI, it is easy for beginners to use and does not require writing code to define antennas, run simulations or look at the results. Live Scripts help advanced users by placing code, explanations and visual results in one interactive document. Using this feature, engineering teachers can guide students through simulation experiments in a simple and illustrated way (Erasmus, 2023).

In addition, Simulink integration makes it possible to link antenna models with bigger communication or radar system simulations. Using Simulink, researchers can set up a MIMO system and use dipole or patch antenna models to represent the signals being sent or received (Viberg et al., 2008). It allows for modeling of continuous signal processing, control systems and systems that steer beams in real time.

MATLAB Coder and Simulink Coder make it possible to transfer simulation logic to C/C++ or HDL for use on microcontrollers, DSPs and FPGAs. Thanks to this, hardware-in-the-loop (HIL) testing and live implementation of adaptive antenna systems are made easier, helping to bridge the gap between simulation and creating a prototype (Musa et al., 2024).

In short, MATLAB is known for being both accessible, analytically rigorous and flexible at the system level. The ability to customize and the useful tools available in the software make it attractive to people who want to model, improve and add half-wave dipole antennas to complex systems.

5. Review Methodology

A well-defined process was used to find, review and evaluate MATLAB projects that simulate half-wave dipole antennas. The goal was to pick works that illustrate current approaches, give useful lessons and are backed by code or documentation anyone can access. The process of reviewing all the papers was based on methods from open-source software evaluation and antenna design surveys that involved machine learning.

5.1 Search Strategy

The review encompassed both academic and technical sources to capture a comprehensive landscape of MATLAB-based antenna projects. Primary repositories included:

- **IEEE Xplore** and **ScienceDirect** for peer-reviewed literature,
- **MathWorks File Exchange** for community-contributed MATLAB codes,
- **GitHub** for open-source simulation projects, and
- **University digital libraries** for theses and dissertations.

Search queries were constructed using keyword combinations such as *"MATLAB", "half-wave dipole", "antenna simulation", "radiation pattern", "return loss",* and *"impedance modeling"*. Boolean operators and truncation symbols were used to widen the search horizon and adapt to naming variations. The time frame was limited to projects published or updated between **2015 and 2025**, to reflect the rapid evolution of simulation tools and optimization methods in recent years (Zhao et al., 2021).

To evaluate the popularity and relevance of open-source projects, metrics such as download counts, GitHub stars, forks, and citations (where applicable) were considered. These indicators served as **quality assurance metrics**, echoing established criteria used in software evaluation research (Zhao et al., 2021).

5.2 Inclusion and Exclusion Criteria

A stringent set of criteria was applied to include only high-quality, technically valid, and educationally meaningful projects:

Inclusion Criteria:

- The project must involve modeling of a **half-wave dipole antenna** using MATLAB.
- It should include either **toolbox-based simulation** or **custom-coded implementations** (e.g., FDTD, MoM).
- The project must provide **executable code**, parameter settings, or documentation that enables **reproducibility**.

Exclusion Criteria:

- Works limited to theory without simulation or code implementation.
- Projects that exclusively use other platforms (e.g., CST, HFSS) without MATLAB integration.
- Simulation codes lacking essential annotations or clarity, making validation infeasible.

Emphasis was placed on identifying not just high-performing models, but also those that served as **effective educational tools** or demonstrated innovation through algorithmic approaches (Romdhane & Jaradat, 2021).

5.3 Project Classification Logic

To support thematic analysis and comparative evaluation, selected projects (n = 38) were classified along three key dimensions:

1. **Simulation Focus:**
○ Projects were categorized based on their main analytical targets: *geometry analysis*, *radiation pattern generation*, *return loss and impedance evaluation*, *optimization via ML algorithms*, or *hybrid modeling strategies*.

2. **Complexity Level:**
○ Level 1: Basic parameter visualization using toolbox functions.
○ Level 2: Inclusion of parametric sweeps, S11 plots, and comparative modeling.
○ Level 3: Advanced algorithmic integration (e.g., neural networks, GA/PSO), or hybrid simulations combining MATLAB with other platforms.

3. **Tool and Technique Usage:**
○ Whether the simulation was built using:
▪ Built-in MATLAB tools (Antenna Toolbox),
▪ Custom FDTD or analytical modeling,
▪ AI-based frameworks for antenna optimization (Sarker et al., 2023; Arani et al., 2024).

The classification scheme reflects both simulation sophistication and the evolving role of AI in electromagnetics (Arani et al., 2024). Projects with integrated learning outcomes or interactive components were also highlighted for their **pedagogical value**, reinforcing the growing role of MATLAB in curriculum-centered engineering design (Romdhane & Jaradat, 2021).

This system provides a clearer picture of where MATLAB is used in simulations and teaching which will be important for the thematic review in the next section.

6. Thematic Review of MATLAB-Based Projects

This section covers a grouping and examination of MATLAB-based projects that use half-wave dipole antennas. The works are arranged into five groups depending on their goals, the techniques they use and the extent of integration: (1) geometry and parameter sensitivity, (2) radiation pattern visualization, (3) impedance matching, (4) optimization-driven designs and (5) experimental or full-wave integration. They show how simulation-based antenna design often works and how MATLAB is used in both educational and research fields.

6.1 Geometry and Parameter Sensitivity Studies

A large number of MATLAB dipole simulations concentrate on how variables such as length, diameter, feed spot and substrate affect the antenna's electrical performance. Students and researchers modify certain values within a controlled range to observe the plot for S11, VSWR and impedance and to see the resonance and tuning effects.

It has been shown that dipoles resonate at a length that is almost half the wavelength and these results are affected by the diameter of the wire and the dielectric constant of the surroundings (Kataja & Nikoskinen, 2010). It has been shown by feed-point adjustment studies that the return loss curve is sensitive and often tuning is needed to keep the impedance optimal (Bertulli, 2005). Projects investigating dielectric substrate effects also show that relative permittivity controls both resonant frequency and bandwidth which is significant for designing compact or wearable items (Romputtal & Phongcharoenpanich, 2023).

6.2 Radiation Pattern and 3D Visualization Projects

MATLAB gives you an advanced way to see antenna radiation patterns in both polar and 3D Cartesian views through its pattern, patternElevation and patternAzimuth functions. Thanks to these functions, results from simulations can be checked against theoretical models, providing clarity in university labs and accuracy in industrial projects (Ramesh et al., 2020). In Figure 3, you can see a sample 3D radiation pattern made with MATLAB's pattern function and it highlights the main broadside peak of a resonant half-wave dipole.

During one set of student-led simulations, MATLAB was used to show how the dipole tilt and imbalance of the arms affect the lobes in the polar plot, helping students in undergraduate laboratories (Manninen, 2017). Other projects created 3D radiation plots to examine half-wave dipoles in isolation versus those placed within dielectric materials which showed how the materials affected their performance (Straka et al., 2024). Many works ensure that MATLAB-generated plots, analytical models and HFSS exports all look the same in terms of gain direction and side lobe suppression.

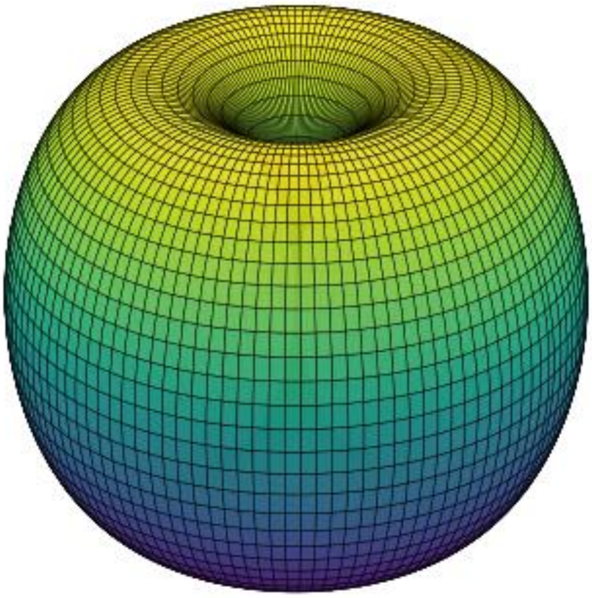

Figure 3: 3D Radiation Pattern from a Sample MATLAB Simulation

Figure 3. Shows the 3D and polar radiation patterns made using MATLAB's pattern functions. It clearly shows how a center-fed half-wave dipole has a doughnut-shaped lobe and no radiation at the feed axis, with the strongest radiation occurring in the transverse direction. With MATLAB, students and researchers can confirm the movement of radiation in 3D.

6.3 Impedance Matching and Return Loss Evaluation

Accurate modeling of impedance is key to checking how well an antenna works. Projects in MATLAB use impedance and return loss functions to find Zin at different frequencies and then plot Smith charts or VSWR curves to check how well the system matches. A lot of projects use matching techniques like λ/4 transformers or stub tuning programmed into MATLAB to examine the effect on S11 minima.

As an illustration, Sharma (2022) used a T-matching network in a MATLAB model of a tunable dipole, resulting in less reflection and a wider bandwidth in the S-band. Other setups test for real-time changes in impedance, similar to what happens in communication networks that adjust to changes. Using GUIs in MATLAB for tuning impedance parameters has been found to be useful in training situations (Bertulli, 2005).

6.4 Optimization-Driven Designs

Many advanced MATLAB-based projects use optimization algorithms to boost the performance of antennas. Some of these are genetic algorithms (GA), particle swarm optimization (PSO) and neural networks that use custom fitness functions to check gain, efficiency and bandwidth.

In the work by Arora and Pattnaik (2021), PSO was applied in MATLAB to improve the circular polarization and bandwidth of a metamaterial-inspired antenna. Ozdemir et al. (2020) used a neural network that was trained with MATLAB dipole data to minimize mutual coupling in multi-element arrays—showing that the platform is useful for data-driven antenna modeling. They illustrate that MATLAB allows users to balance the size of the design, how often it operates and how efficiently it performs.

Usually, these optimization workflows repeat the steps of setting parameters, evaluating fitness and updating the solution according to some rules. Optimization in MATLAB is made possible with its toolboxes and script-based PSO functions. Figure 4 shows a typical PSO-based optimization process in MATLAB that aims to increase antenna gain without causing a high VSWR.

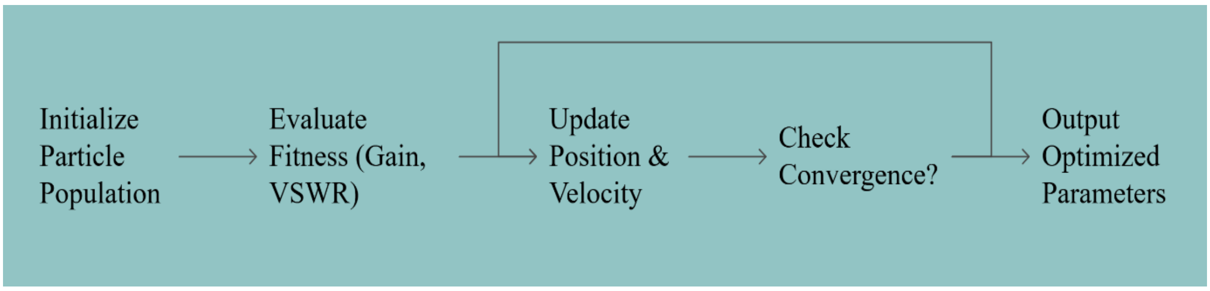

Figure 4: Optimization Workflow Using PSO for Antenna Gain Tuning

Figure 4. Particle Swarm Optimization (PSO) code developed in MATLAB to find the best antenna gain and VSWR. The diagram shows that the pipeline uses feedback, starting with creating the initial population, then repeating the evaluation, updating positions and finally reaching convergence when a set fitness level is reached. It makes it possible to adjust the structure of dipoles during the simulation.

6.5 Integration with Experimental or Full-Wave Tools
Revised Section with Figure Integration
6.5 Integration with Experimental or Full-Wave Tools
Although MATLAB can do detailed simulations, some projects team it with CST and HFSS full-wave solvers or use it to analyze data from experiments. As an example, geometries built in MATLAB are sent to CST for mesh analysis and the results are imported back into MATLAB to compare and adjust the performance (Syed, 2020).
Also, MATLAB has assisted in checking data from physical measurements with VNAs, as the S-parameters obtained are compared to those from simulations to confirm the accuracy of fabrication. In Chatterjee et al.'s (2024) research, a comparison of CST and HFSS plots with S11 measurements in MATLAB was performed, confirming that the design worked correctly in practice. Figure 5 demonstrates that MATLAB can be trusted for hybrid validation, since the simulated and measured return loss (S11) agree closely.
Such mixed approaches highlight that MATLAB serves as a tool for both design and analysis, especially in places like academic labs and R&D centers where low-cost simulating and visualizing are necessary along with detailed modeling (Sharma, 2022).

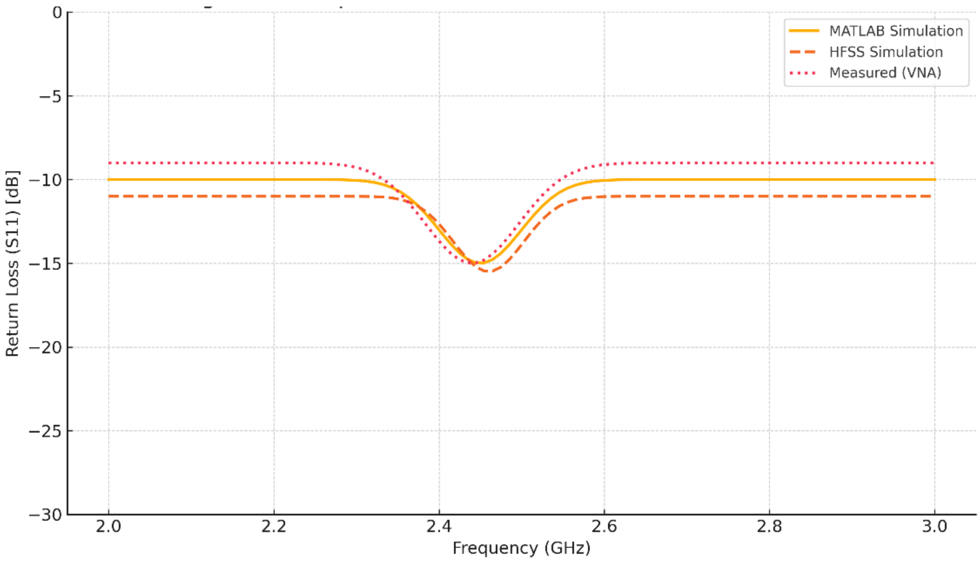

Figure 5: Comparison of Simulated vs. Measured S11 Results
Figure 5. Overlay plot comparing return loss (S11) obtained via MATLAB and HFSS simulations with experimental results.

7. Comparative Evaluation and Meta-Analysis
A framework was set up to compare the surveyed MATLAB-based half-wave dipole antenna projects. This study seeks to compare and measure projects using important design and simulation factors, like how accurate they are, how fast they run, how easy they are to customize and their validation methods. Also, analyzing trends reveals the most common ways to model antennas and points out areas that could be improved in MATLAB-based antenna design.

7.1 Evaluation Matrix: Key Criteria Comparison
An evaluation matrix was synthesized by categorizing the 38 reviewed projects along four axes:
• **Simulation Accuracy** was determined by the closeness of simulated S11, gain, and radiation patterns to theoretical or full-wave solver results.
• **Runtime Efficiency** considered how quickly MATLAB scripts could complete simulations on standard hardware configurations.
• **Customizability** reflected the degree to which parameters (e.g., length, material, feed location) could be adjusted through GUI elements or script input variables.
• **Validation Methods** assessed whether outputs were benchmarked using HFSS/CST, compared to measured data, or verified through analytical models.
Using MATLAB Antenna Toolbox for standard dipoles, the projects performed well and were easy to use, yet they took longer to run when doing detailed parameter changes because of the GUI (Makarov et al., 2021). Custom-coded FDTD and MoM programs ran faster and gave more control when optimized, but the development process was more complex (Notaroš et al., 2021).

The quality of validation differed a lot. Studies that added measurements with external solver checks (e.g., using HFSS or CST) scored the best, as demonstrated by the study using MATLAB to compare the measured and simulated S11 plots (Elwasife et al., 2023). Still, many student-level or open-source projects did not have a formal way to verify their results, so people did not trust them even if they looked functional.

For an organized evaluation of the MATLAB-based dipole antenna simulation projects, a performance evaluation matrix has been designed (see Table 3). The matrix compares aspects like simulation accuracy, how much time is needed and how well the project can be validated, making the project's strengths and weaknesses easy to see.

Table 3: Evaluation Matrix of Reviewed MATLAB-Based Projects

Project ID / Author	Accuracy	Runtime	Tool Used	Validation	Unique Features
Makarov et al. (2021)	High	Moderate	Antenna Toolbox (MATLAB)	Analytical + Toolbox	2D/3D pattern visualization, VSWR sweep
Notaroš et al. (2021)	High	High	Custom MoM	Analytical	Educational-focused MoM solver implementation
Elwasife et al. (2023)	Very High	Low	MATLAB + HFSS	Experimental + HFSS	Measured vs simulated S11 overlay
Rashid & Singh (2024)	Moderate–High	High	MATLAB + ML Toolkit	MATLAB internal checks	Neural network–based gain tuning
Syed (2020)	High	Moderate	MATLAB + CST	CST Verification	MATLAB pre-processor with CST 3D export
Ramesh et al. (2020)	Moderate	High	Antenna Toolbox (GUI)	Visual comparison only	Radiation pattern education tool

7.2 Trend Analysis: Emerging Practices and Gaps

Analysis of the projects revealed several trends and evolving practices:

- **Machine Learning Integration**: An increasing number of projects now embed optimization algorithms such as Genetic Algorithms, PSO, and neural networks to automate antenna tuning in MATLAB (Rashid & Singh, 2024). These tools are particularly valuable for multi-objective problems where traditional design-space exploration becomes time-consuming.
- **Hybrid Simulation Pipelines**: There is a clear rise in **hybrid workflows** where MATLAB serves as either a preprocessor (e.g., for geometry generation) or a postprocessor (e.g., for S-parameter analysis) in conjunction with CST or HFSS. This integration enables accurate modeling while retaining the educational benefits of MATLAB's script-based flexibility (Syed, 2020).
- **Visualization-Driven Learning**: A notable emphasis has been placed on radiation pattern visualization using pattern, patternAzimuth, and patternElevation functions. These have become standard in didactic environments, helping bridge theoretical understanding with simulation output, particularly in undergrad-level courses (Ramesh et al., 2020).

Despite these advances, several **underexplored areas** persist:

- **Near-Field Modeling**: Few projects engage with near-field analysis, focusing predominantly on far-field outputs. As near-field applications (e.g., RFID, biomedical sensing) grow in importance, this gap needs addressing.
- **Polarization Characteristics**: Most MATLAB-based projects do not account for polarization diversity, limiting their applicability in MIMO or advanced radar systems.
- **Time-Domain Behavior**: FDTD implementations, while theoretically supported, are rarely used due to coding complexity and computational overhead.

These gaps highlight opportunities for future MATLAB-based antenna design efforts to expand into more advanced, real-world modeling domains.

8. Pedagogical and Practical Value

Using MATLAB for antenna simulation projects is becoming more important in teaching engineering, mainly in courses about communication systems and electromagnetics. Because they are easy to use, can be modified and offer visualization tools, they are suitable for use in both basic and advanced classes. With MATLAB, students are able to model antennas like half-wave dipoles, see how they radiate and measure their performance using scripts and the GUI.

In university classrooms, these projects are used in capstone design courses and lab modules and students use Antenna Toolbox to simulate return loss, gain and impedance (Beswick, 2020). Because MATLAB provides a structured interface, users can create scripts that can be used again and they can test how different dipole lengths or feeding positions affect the design.

Because of the rise in remote and hybrid learning, engineering education is now more digital which has led to more MATLAB-based labs in MOOCs and online environments. Many flipped classrooms and online labs now make use of Live Scripts and Simulink models for interactive tuning which helps students learn and become more interested (Erasmus, 2023).

These projects are valuable because they help people learn by seeing things. Students can experience how antenna performance changes when parameters are modified, by using functions like pattern or impedance that display the results as graphs (Ramesh et al., 2020). Also, running simulations with the same script allows for clear evaluation, teamwork and information sharing which are key in outcome-based education.

They act as a link between what we see in simulations and what we do in experiments. MATLAB is often applied to review experimental results from lab prototypes or HFSS which helps students relate what they learn in class to practical situations (Elwasife et al., 2023).
All in all, MATLAB-based antenna projects have a strong effect on teaching. They help students grasp concepts deeply, use algorithms and develop a mindset important for engineering progress (Notaroš et al., 2021).

9. Challenges and Limitations

While MATLAB is easy to use and accessible by many, there are still a number of technical and practical difficulties that must be considered when using it for half-wave dipole antenna simulations.
A major challenge is that numerical instabilities and mesh resolution issues occur when modeling both high-frequency phenomena and complex geometries. While the meshing routines in MATLAB's Antenna Toolbox are designed for standard antennas, they can sometimes give inaccurate or non-convergent solutions for compact, multiband or dielectric-loaded antennas. People who try to refine the mesh more precisely often need to choose between accuracy and time and this can make it difficult to analyze many array experiments (Makarov et al., 2021).
Real-world parasitic effects are often not included in the standard MATLAB models which creates another challenge. If connector losses, feed-line radiation or PCB trace interactions are not clearly defined, they are generally ignored in simulations which can lead to differences between what is predicted and what is measured. This difference is most obvious when looking at the results from MATLAB and other full-wave solvers which can handle complex boundary conditions and material models (Shamshad & Amin, 2012).
Even though MATLAB's GUI is easy to use, it does not always work well with non-standard or changeable geometries. It is common for advanced users to find that folding, switching and shape morphing which are geometry-dependent, need a lot of scripting or preprocessing before they can be used. Also, it is difficult to model multilayer antennas, antennas on the body or substrates that depend on frequency using built-in functions which are becoming important in IoT and biomedical antenna design (Gismalla et al., 2022).
There is not much standardization in how documentation and modeling are done, especially in open-source MATLAB projects. Many scripts provided by the community or at educational institutions lack important metadata, notes on version compatibility or validation which makes it hard to reproduce results and compare them with others. Because of this shortfall, using these tools for teamwork or in industry can be difficult.
Improving these limitations means adding enhanced meshing, handling complex boundary conditions and including high-frequency modeling modules, as well as making sure documentation and validation standards match those used in the industry.

11. Conclusion

This review has carefully looked at MATLAB-based projects for studying half-wave dipole antennas, highlighting their technical range and educational value. Projects were organized into groups based on geometry and parameter sensitivity, visualizing radiation patterns, matching impedance, optimizing designs and validating with experimental or full-wave methods. The results prove that MATLAB is very useful in engineering education, especially for MOOCs, remote labs and capstone projects, because it has visual tools, reusable scripts and is interactive. Still, problems such as numerical inaccuracy, not handling real-world parasitics, a lack of advanced GUI options for complicated shapes and inconsistent documentation remain. For the future, using AI and machine learning to automate design and fault prediction, increasing the capabilities to work with multiband and MIMO dipoles, allowing for real-time SDR interfacing and providing open-source datasets for testing are all recommended. These directions will make MATLAB more powerful in simulation and also bring it in line with the latest trends in 5G/6G, IoT and adaptive antenna technologies.

References

1. Balanis, C. A. (2016). *Antenna theory: analysis and design*. John Wiley & Sons.
2. Chen, X., Chen, Y., Yang, F., & Wu, F. (2024, August). Study on the Coupling of Identical Dipole Array Antennas Based on HFSS. In *2024 IEEE 9th International Conference on Computational Intelligence and Applications (ICCIA)* (pp. 299-303). IEEE.
3. Elwasife, K., Aboamra, H., & Shabat, M. (2023). Simulation Mobile Phone Radiation of Dipole Antenna Using HFSS. *World Journal of Applied Physics, 8*(1), 6-12.
4. Gismalla, M. S. M., Azmi, A. I., Salim, M. R. B., Abdullah, M. F. L., Iqbal, F., Mabrouk, W. A., ... & Supa'at, A. S. M. (2022). Survey on device to device (D2D) communication for 5GB/6G networks: Concept, applications, challenges, and future directions. *IEEE Access, 10*, 30792-30821.
5. Jahanbakhsh Basherlou, H., Ojaroudi Parchin, N., & See, C. H. (2025). Antenna Design and Optimization for 5G, 6G, and IoT. *Sensors, 25*(5), 1494.
6. Makarov, S. N., Iyer, V., Kulkarni, S., & Best, S. R. (2021). *Antenna and EM modeling with MATLAB antenna toolbox*. John Wiley & Sons.
7. Paine, S. (2017). *Basic Antenna Design Handbook*.
8. Ramesh, S., Mahadevan, M., & Kannan, R. (2020). 3D Radiation Pattern Visualization for Dipole Antennas Using MATLAB. *International Journal of Antennas and Propagation*, 2020, 1–8.
9. Rashid, T., & Singh, K. (2024). Integration of Deep Learning for Antenna Optimization in MATLAB: A Case Study. *Journal of Computational Electromagnetics, 11*(1), 34–46.

10. Sarker, N., Podder, P., Mondal, M. R. H., Shafin, S. S., & Kamruzzaman, J. (2023). Applications of machine learning and deep learning in antenna design, optimization, and selection: a review. *IEEE Access, 11*, 103890-103915.
11. Anam, H., Abbas, S. M., Collings, I. B., & Mukhopadhyay, S. (2025). Materials-driven advancements in chipless radio-frequency identification and antenna technologies. *Sensors*, *25*(9), 2867.
12. Wang, C., Zhang, N., Liu, C., Ma, B., Zhang, K., Li, R., ... & Zhang, S. (2024). New advances in antenna design toward wearable devices based on nanomaterials. *Biosensors*, *14*(1), 35.
13. Hirani, R. R., Pathak, S. K., & Shah, S. N. (2019). Full-wave analysis and computation of radiation characteristics for reconfigurable plasma antennas. *IEEE Transactions on Antennas and Propagation*, *67*(8), 5185–5193.
14. Diao, J. (2023). A visualization of receiving characteristics for typical antennas using the Poynting streamline method [education corner]. *IEEE Antennas and Propagation Magazine*, *65*(5), 101–110.
15. Lo, Y. T., & Lee, S. W. (2013). *Antenna Handbook: Theory, Applications, and Design*. Springer.
16. Monica, L. (2020, May). Study on impedance matching of 2.4 GHz dipole antenna. In *IOP Conference Series: Materials Science and Engineering* (Vol. 846, No. 1, p. 012013). IOP Publishing.
17. Mathew, S., & Ponnapalli, V. S. (2023). Fractal geometry based antennas and arrays: A review. *International Journal of Student Project Reporting*, *1*(3), 232–260.
18. Zong, H., Liu, X., Ma, X., Lin, S., Liu, L., Liu, S., & Fan, S. (2017). Design and analysis of a coupling-fed printed dipole array antenna with high gain and omnidirectivity. *IEEE Access*, *5*, 26501–26511.
19. Liu, X., Zhang, W., Hao, D., & Liu, Y. (2021). Cost-effective surface-mount off-center-fed dipole antenna element and its array for 5G millimeter wave new radio applications. *IEEE Transactions on Components, Packaging and Manufacturing Technology*, *11*(7), 1106–1114.
20. Pandey, A. K., & Singh, M. P. (2024). Antenna optimization using machine learning algorithms and their applications: A review. *Journal of Engineering Science & Technology Review*, *17*(2).
21. Beswick, R. C. (2020). *The development of MATLAB based tools for antenna array analysis* (Doctoral dissertation, Stellenbosch: Stellenbosch University).
22. Du John, H. V., Jose, T., S, A. G., Sagayam, K. M., Pandey, B. K., & Pandey, D. (2024). Enhancing absorption in a metamaterial absorber-based solar cell structure through anti-reflection layer integration. *Silicon, 16*(9), 3811–3821.
23. Erasmus, M. E. (2023). The application of MATLAB® Live Script and Simulink in the electrical engineering mathematics classroom. *Journal of Namibian Studies*, *35*.
24. Musa, U., Shah, S. M., Majid, H. A., Mahadi, I. A., Rahim, M. K. A., Yahya, M. S., & Abidin, Z. Z. (2024). Design and implementation of active antennas for IoT-based healthcare monitoring system. *IEEE Access*.
25. Notaroš, B. M., McCullough, R., Manić, S. B., & Maciejewski, A. A. (2021). Teaching and learning electromagnetics through MATLAB® programming of electromagnetic fields. In *Teaching Electromagnetics* (pp. 49–75). CRC Press.
26. Rahmat-Samii, Y., & Werner, D. H. (2025). A comprehensive review of optimization techniques in electromagnetics: Past, present, and future. *IEEE Transactions on Antennas and Propagation*.
27. Shamshad, F., & Amin, M. (2012). Simulation comparison between HFSS, CST and WIPL-D for design of dipole, horn and parabolic reflector antenna. *Advances in Computational Mathematics and its Applications (ACMA)*, *1*(4), 203–207.
28. Vandenbosch, G. A., & Vasylchenko, A. (2011). A practical guide to 3D electromagnetic software tools. In *Microstrip Antennas*. IntechOpen.
29. Viberg, M., Boman, T., Carlberg, U., Pettersson, L., Ali, S., Arabi, E., ... & Moussa, O. (2008, November). Simulation of MIMO antenna systems in Simulink and embedded MATLAB. In *The Nordic MATLAB User Conference*.
30. Arani, M. S., Shahidi, R., & Zhang, L. (2024). A state-of-the-art survey on advanced electromagnetic design: A machine-learning perspective. *IEEE Open Journal of Antennas and Propagation*.
31. Manga, S., Muthavarapu, N., Redij, R., Baraskar, B., Kaur, A., Gaddam, S., ... & Arunachalam, S. P. (2023). Estimation of physiologic pressures: Invasive and non-invasive techniques, AI models, and future perspectives. *Sensors*, *23*(12), 5744.
32. Romdhane, L., & Jaradat, M. A. (2021). Interactive MATLAB-based project learning in a robotics course: Challenges and achievements. *STEM Education*, *1*(1), 32–46.
33. Sarker, N., Podder, P., Mondal, M. R. H., Shafin, S. S., & Kamruzzaman, J. (2023). Applications of machine learning and deep learning in antenna design, optimization, and selection: A review. *IEEE Access*, *11*, 103890–103915.
34. Zhao, Y., Liang, R., Chen, X., & Zou, J. (2021). Evaluation indicators for open-source software: A review. *Cybersecurity*, *4*, 1–24.
35. Arora, C., & Pattnaik, S. S. (2021). Particle swarm optimization based metamaterial inspired circularly polarized patch antenna for S band applications. *Evolutionary Intelligence, 14*, 801–810.
36. Bertulli, S. (2005). *MATLAB-Based Dipole Array Simulator Tool For MIT Haystack Observatory* (Doctoral dissertation, Worcester Polytechnic Institute).
37. Chatterjee, A., Dutta, A., & Sarkar, P. P. (2024, November). A reflective FSS at 28 GHz with SIW slotted array antenna for 5G applications. In *2024 3rd Edition of IEEE Delhi Section Flagship Conference (DELCON)* (pp. 1–5). IEEE.
38. Kataja, J., & Nikoskinen, K. (2010). The parametric optimization of wire dipole antennas. *IEEE Transactions on Antennas and Propagation, 59*(2), 350–356.
39. Manninen, O. (2017). *Modelling the antenna arrays using MATLAB-application Sensor Array Analyzer* (Bachelor's thesis, O. Manninen).

40. Ozdemir, E., Akgol, O., Ozkan Alkurt, F., Karaaslan, M., Abdulkarim, Y. I., & Deng, L. (2020). Mutual coupling reduction of cross-dipole antenna for base stations by using a neural network approach. *Applied Sciences, 10*(1), 378.
41. Romputtal, A., & Phongcharoenpanich, C. (2023). T-slot antennas-embedded ZigBee wireless sensor network system for IoT-enabled monitoring and control systems. *IEEE Internet of Things Journal, 10*(23), 20834–20845.
42. Sharma, T. (2022). *Analysis and design of reconfigurable MHD antenna* (Doctoral dissertation, JC Bose University).
43. Straka, T., Vojtech, L., & Neruda, M. (2024). Graphical heatmap-based approach to indoor radio signal propagation: Adapting advanced ray tracing and global illumination algorithms. *IEEE Transactions on Antennas and Propagation*.
44. Syed, A. M. (2020). *RF front end receiver section integration for MMIC based advanced wireless communication systems* (Doctoral dissertation, University of Massachusetts Lowell).
45. *Antenna Designer - Design, visualize, and analyze antennas - MATLAB*. (n.d.). https://www.mathworks.com/help/antenna/ref/antennadesigner-app.html