TITLE: AI-Based System for 6G Reconfigurable Intelligent Surfaces

Group Number: 27

CLASS: ECE 492

NAMES: Loukas Athanitis, Brian Cole, Mari Takizala, Surabhi Rajbhandari

ADVISOR: Dr. Dandekar

DATE: 3/14/2025

Abstract

As wireless communication technologies are introduced into more aspects of our lives, from Internet of Things (IoT) to AI computing on the network edge, the number of wireless users and devices is also increasing. To meet this demand, the next generation of wireless communication requires a scalable and sustainable solution to ensure fast and reliable connectivity. Reconfigurable Intelligent Surfaces (RIS) use passive beamforming to reach wireless endpoints and reduce the impact of dead zones. However, the research and development of RIS systems is still in its early stages. Most research remains theoretical, with limited experimental implementations exploring its real-time adaptability. Additionally, prototypes tend to be narrowband, which are not ideal for the large frequency bands comprising 6G and are difficult to scale. We are designing a framework that combines machine learning, channel estimation, ray tracing simulation, and embedded design to tune the antenna elements on the surface in real time. Our approach combines three key components: RIS simulation to model surface behavior to ray trace with different configuration angles, a channel estimation pipeline utilizing synthetic channels from the simulation to evaluate throughput and signal strength, and AI-driven reflection angle optimization using a reinforcement learning model to achieve high signal to noise ratio (SNR). Using a trained Q-learning model, the RIS will configure its reflection angle to maximize throughput for an end user at a given position.

Table of Contents

Table of Figures

Table of Tables

1. Introduction

1.1. Motivation

Urban environments present numerous signal disruptions. Next-generation cell service must excel in both outdoor urban areas and challenging indoor spaces. Current state-of-the-art alternatives rely on active repeaters, which tend to be energy-intensive and can amplify unwanted noise with a signal.

1.2. Problem Statement

We aim to design a Machine Learning (ML) based system to support real-time optimization of a Reconfigurable Intelligent Surface (RIS) to enhance wireless communication performance by increasing throughput and improving signal coverage, providing a significant improvement over scenarios without RIS implementation.

1.3. Stakeholder Needs

Stakeholders and their customers have needs and with that comes specifications. The system needs to address dead zones present without the RIS. This is done by configuring the board to aim the incoming signal towards the dead zone. An example is shown in the Wireless InSite simulation below in Figure 1-1:

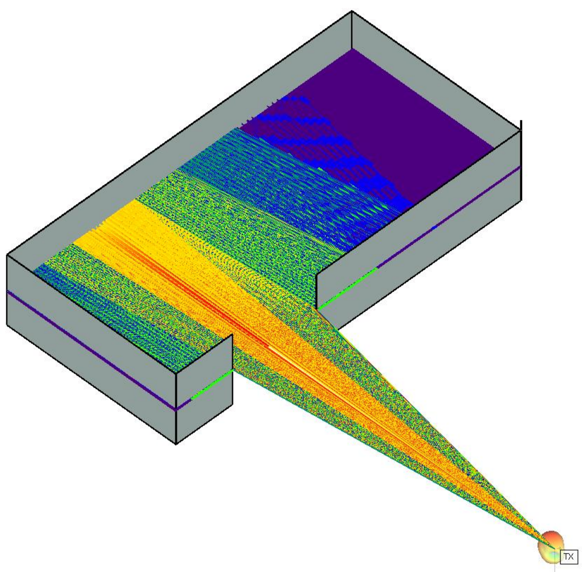


Figure 1-1 Dead zones present without the RIS

On top of expanding coverage, users should be able to expect to receive a relatively strong signal, which requires achieving a high signal-to-noise ratio (SNR). Stakeholders investing in RIS technology would primarily be concerned with ensuring that the technology is effective while remaining sustainable and relatively low cost to manufacture and deploy. The design of the reconfigurable surface is beyond the scope of our project; rather, our system should focus on being energy-efficient, adaptable to different environments, and scalable.

1.4. State of the Art

We have considered smart repeaters as an alternative solution to address the limitations of wireless coverage. [4] outlines the features of RIS technology that delineates it from smart repeaters or MIMO relays, which are amplify and forward (AF) devices. The most important factor that differentiates an RIS system from comparable technology is its passive nature. Where smart repeaters actively amplify the power of a signal, the RIS can passively redirect signals by improving the channel between transmitter and receiver via its surface configuration. RIS will operate with lower power consumption and a better signal-to-noise ratio since it does not actively amplify the power of the incident signal, as well as potentially amplifying noise, with relatively minimal power requirements. Active RIS implementations have been proposed, such as the RIS-based information transmission approach in [8], but we have prioritized passivity in our design. At scale, active RIS systems raise concerns about higher power requirements.

1.5. Approach

We utilized a combination of simulated scenarios and channel estimation. Wireless InSite, a ray-tracing software, was used to analyze received power at specific points in the room. An automation algorithm in Python was developed to efficiently simulate virtually all possible angles for the RIS, generating a dataset that analyzed the impact of different RIS reflection angles on received power at various positions and was used to train an AI model. For verification purposes outside of simulation, we tested real-life wireless channels with custom Software Defined Radio (SDR) designs, starting with a copper sheet as a simple reflector. The qualifying metrics measured for this design are signal-to-noise ratio, error vector magnitude, and bit error rate for transmitted and reflected signals. All data was collected using Drexel Wireless System's Laboratory's DragonRadio SDR image.

1.6. Prototyping Hypothesis/Questions

1.6.1. How do we evaluate the indoor performance of the RIS?

Use Wireless InSite to measure the received power at a specific point in the room. Wireless InSite has a meta surface feature where we can create an RIS and set its reflection angle. This way we can measure the improvements of received power between different refection angles and the improvement of coverage with the presence of an RIS.

1.6.2. How do we train a model to compute the optimal configuration?

We used a reinforcement learning based DQN model. The model was trained using a Wireless InSite Simulation result dataset. This dataset consisted of Receiver positions, and power it received based on different RIS configuration angles. The AI model learns the optimal angles by interacting with the environment. At each receiver position, the model selects RIS reflection, and receives awards or penalties based on how well the chosen angle enhances the received signal power. It refines its decision-making process over multiple training episodes.

1.6.3. What factors enable the system to achieve near real-time configuration?

For reconfigurable surfaces to be a viable technology, they must adapt to the channel conditions between a transmitter and receiver as quickly as possible. The embedded control system, which acts as middleware between the software layer and the surface, should have as little latency as possible. A significant portion of the latency is introduced by the edge TPU as it performs the necessary computations to determine the optimal configuration for the surface. This necessitates a controller design that is able to process the signal and control the surface using the output from the machine learning model with as little delay possible.

2. Materials and Methods

2.1. Hardware

2.1.1. Microcontroller

The microcontroller unit acts as a middleware layer between the Edge AI device and the RIS surface, translating the high-level control signals generated by the Edge AI into precise commands that adjust the configuration of the RIS elements. This ensures seamless communication and synchronization, allowing the RIS to dynamically modulate signals in response to the real-time optimization performed by the Edge AI. Field programmable gate arrays (FPGAs) were another device that was considered, but an STM32 development board was ultimately chosen for the middleware device, as shown in Figure 8-1 in Appendix B.

2.1.2. Edge AI Device

The trained DQN reinforcement learning model will run on the Google Coral Edge with Tensor Processing Unit (TPU), which offers fast and efficient AI processing with low power consumption. The TPU accelerates neural network execution, enabling us to process real-time feedback and adapt RIS configurations with high efficiency. It will communicate the calculated angle optimization directly to the middleware controller as a control signal to reconfigure the surface.

2.1.3. Software Defined Radio Grid (Dragon Radio)

For a software defined radio (SDR)-based testing setup, the Drexel Wireless System's Laboratory's Dragon Radio will be used. The SDR grid will be utilized for transmitting and receiving data to support channel estimation and real-time monitoring in RIS-aided communication links. Channel estimation between the transmitter and receiver enables the system to adjust modulation schemes for optimal transmission. Additionally, feedback from channel estimation allows the embedded controller to maintain signal integrity in real-time.

2.2. Software

2.2.1. Deep Reinforcement Learning

We have chosen Reinforcement Learning (RL), specifically Deep Q-Network (DQN) model to optimize RIS configuration and maximize signal power. Reinforcement Learning is ideal for problems involving complex state-action spaces where the AI model learns from trial-and-error interaction with an environment. The learning process is guided by the Markov Decision Process (MDP) framework, which consists of state (current situation), action (decision the AI agent makes), reward (numerical feedback indicating quality of chosen action), and policy (strategy AI agent follows to decide based on the state). The training dataset will be generated through simulation in Wireless InSite.

2.2.2. ZephyrOS

We chose to use ZephyrOS as a real-time operating system for processing commands from the edge AI device and controlling the antenna elements on the RIS simultaneously. ZephyrOS is a widely adopted open-source platform that is lightweight, flexible, and has extensive hardware support. It is a reliable choice to handle data flow and task scheduling on the controller unit. The ZephyrOS framework supports applications written in C across a wide variety of microcontrollers.

2.2.3. Ansys HFSS

This software is especially suitable for simulating and analyzing the electromagnetic performance of the RIS board. Specifically, we will use HFSS to confirm that the RIS element configuration achieves the desired beamforming by controlling the phase of reflected waves. HFSS will also be used to generate far-field plots, such as radiation patterns, which are critical for evaluating the RIS's ability to passively control and redirect electromagnetic waves effectively.

2.2.4. Wireless InSite

Wireless InSite will be used to evaluate the performance of the RIS within a realistic environment. This software enables us to model complex environments, including obstacles like buildings and walls, to analyze how the RIS improves wireless coverage and mitigates signal degradation. For instance, Figure 3-3 shows a Wireless InSite simulation where the RIS significantly enhances coverage in previously shadowed regions. Wireless InSite also calculates critical performance metrics such as received power and signal-to-noise ratio (SNR). These metrics are essential for quantifying the benefits of integrating the RIS, including increased signal reliability, reduced dead zones, and improved throughput without requiring active signal amplification.

2.2.5. MATLAB

MATLAB plays a key role in the digital signal processing development aspect of this project. It will mainly be used to prototype channel estimation algorithms by leveraging its various simulation and signal processing toolboxes. These algorithms will be developed to analyze and predict the behavior of communication channels, ensuring that transmissions that the RIS surfaces, and controller detects perform optimally and reliably under various conditions. This will also further support real-world testing, allowing algorithms to be deployed and evaluated on hardware platforms.

2.2.6. Python

Python plays a key role in AI Model Development using frameworks like TensorFlow and PyTorch to optimize RIS configurations based on real-time feedback. It will also be used to automate the iterative process of modifying reflection angles, running continuous simulations in Wireless InSite, and analyzing channel coefficients to calculate SNR. It also bridges AI models with ZephyrOS, enabling real-time RIS configuration adjustments.

3. Results

3.1. Simulation

To simulate the indoor performance of the RIS, we need to create an indoor environment that would showcase the capabilities of the RIS. We picked a “L” shaped room so that there would be significant dead zones in the bent portion of the room. The transmitting antenna is outside, across from the opening of the room to simulate the signal coming from a base station or a similar signal source. The images below in Figure 3-1 and Figure 3-2 shows the results of the simulation without the RIS:

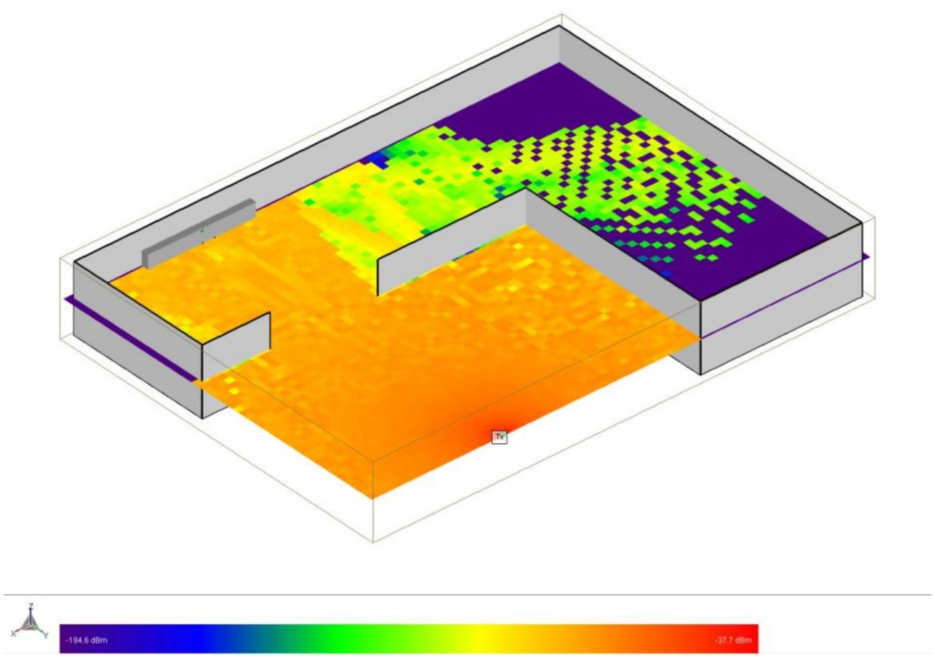


Figure 3-1. Isometric View of Normal Coverage Simulation

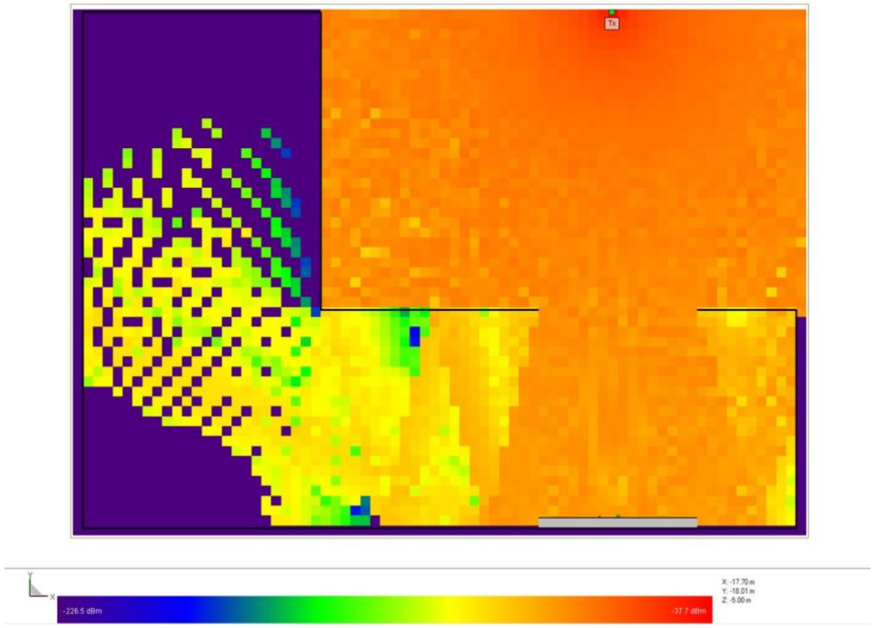


Figure 3-2. Aerial View of Normal Coverage Simulation

The dead zones are illustrated with the color corresponding to the left most side of the spectrum shown below the heatmap. The dead zones on the left side of the room are significant. If a user was in any of the dark zones, they would get little to no coverage. The image that follows shows the simulation results with an RIS present and reflection angle pointed towards the dead zones. Views of these results can be observed in Figure 3-3 and Figure 3-4.

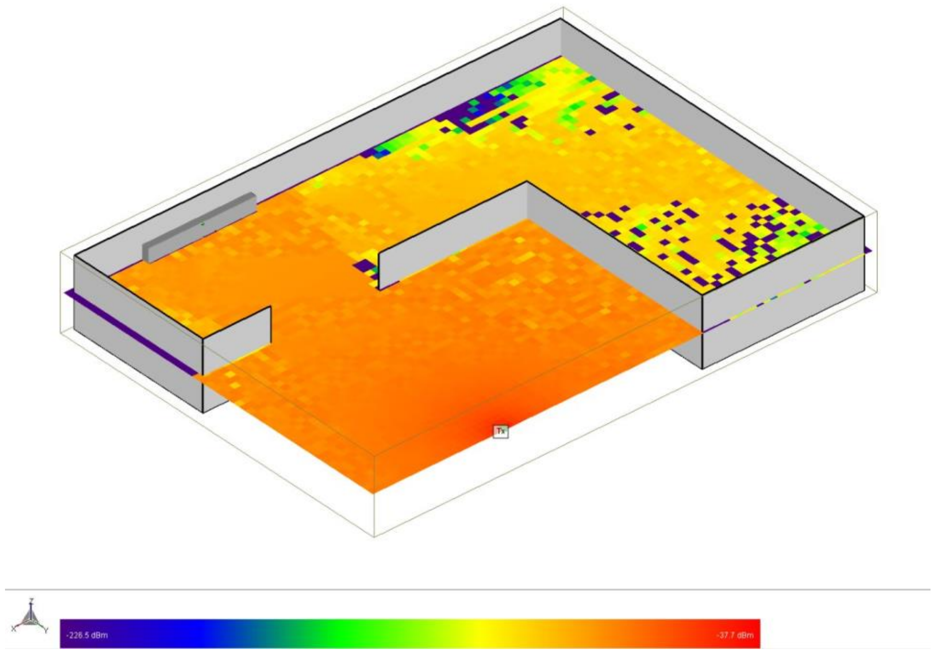


Figure 3-3. Isometric View of Improved Coverage Simulation

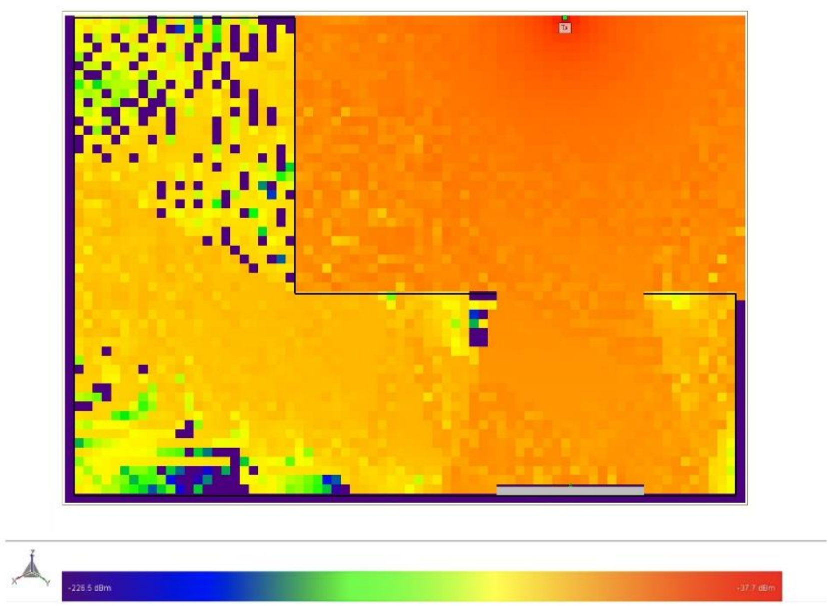


Figure 3-4. Aerial View of Improved Coverage Simulation

As shown above, the RIS improves coverage by delivering signal towards dead zones. It may be worth noting that there are little pockets of dead zones that remain. The RIS is designed to be able to change its reflection angles. So, for example, if a user was in one of those pockets of low coverage, this RIS angle is not optimal for that user location. This simulation model is what was used to train the AI model so that the RIS configuration could be optimized with minimum latency.

3.2. AI Model

We implemented a reinforcement learning based Deep Q-Network (DQN) model to optimize RIS configuration. The state of the system is defined by key parameters such as receiver's position (X, Y coordinates), RIS configuration angles, and received power at receiver's position for different angles.

To train the model, we used a dataset generated though Wireless InSite simulations. We developed a script to automate the simulation process where the RIS was configured for each angle from –170° to 170°. The script collected received power values at different receiver positions, creating a comprehensive dataset for AI model training.

The AI agent interacts with the environment by selecting an RIS reflection angle from the predefined range. It explores different angles to determine the optimal angle that enhances signal power. The agent receives awards or penalties based on how well the chosen angle enhances received signal power. The reward function is directly based on the received power level, meaning higher power values result in higher rewards. This guides the AI agent towards optimal angle selection. Through iterative learning and experience, the model refines its decision-making process to optimize reflection angles and maximize received power at different receiver positions.

The results of using Deep Q-Network (DQN) algorithm demonstrated a good correlation between predicted and optimal RIS reflection angles with high R^2 score (0.979) and low Mean Absolute Error (5.316). The graph and model evaluation can be seen below in Figure 3-5:

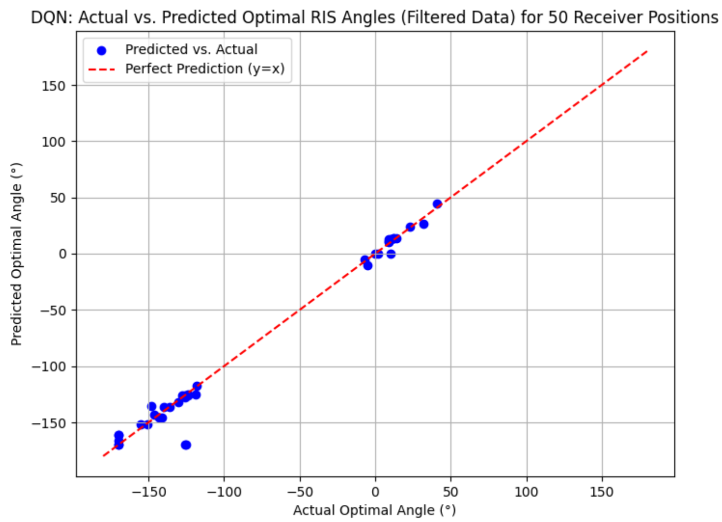


Figure 3-5. Deep-Q Network Model Evaluation Chart

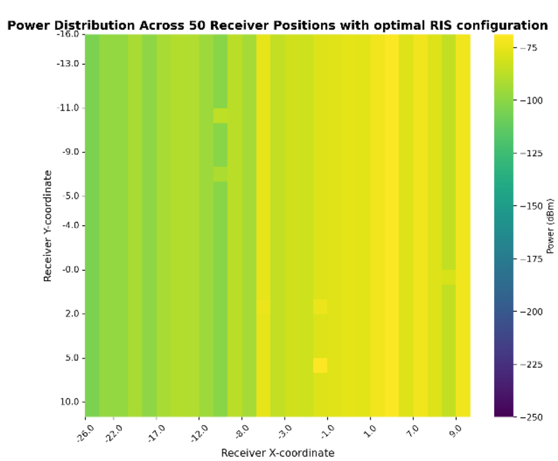
Model Evaluation Metrics (Deep-Q Network):

MAE (Mean Absolute Error): 5.3°

RMSE (Root Mean Squared Error): 11.1°

R^2 Score: 0.979

The heat-maps shown below illustrate the effectiveness of the model in predicting optimal RIS reflection angles. In the heatmap showing the optimal RIS configuration, we observe a high-power distribution across the 50 receiver positions. This shows the AI model's effectiveness in predicting optimal RIS reflection angles, leading to enhanced signal power. In contrast, the heatmap without optimal RIS configuration reveals a poor power distribution as certain receiver positions with dead zones highlighted in purple in Figure 3-6.



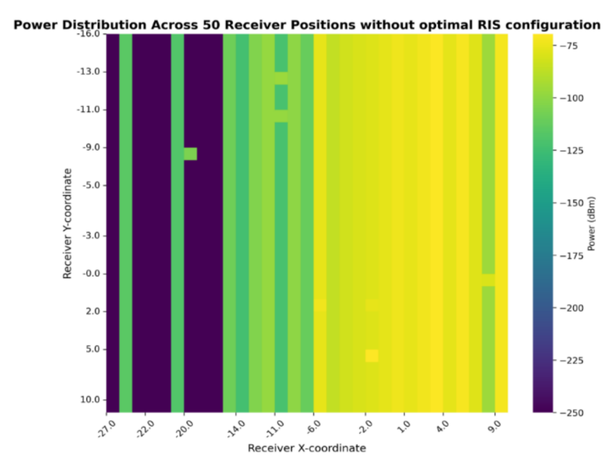


Figure 3-6. Power Distribution Heat-maps for Receiver Positions with and without RIS Configurations

In addition, we also trained two Supervised Learning based models, Random Forest Regression and K-Nearest Neighbors, to predict the optimal RIS reflection angles. The graphs and model evaluations can be seen below in Figure 3-7:

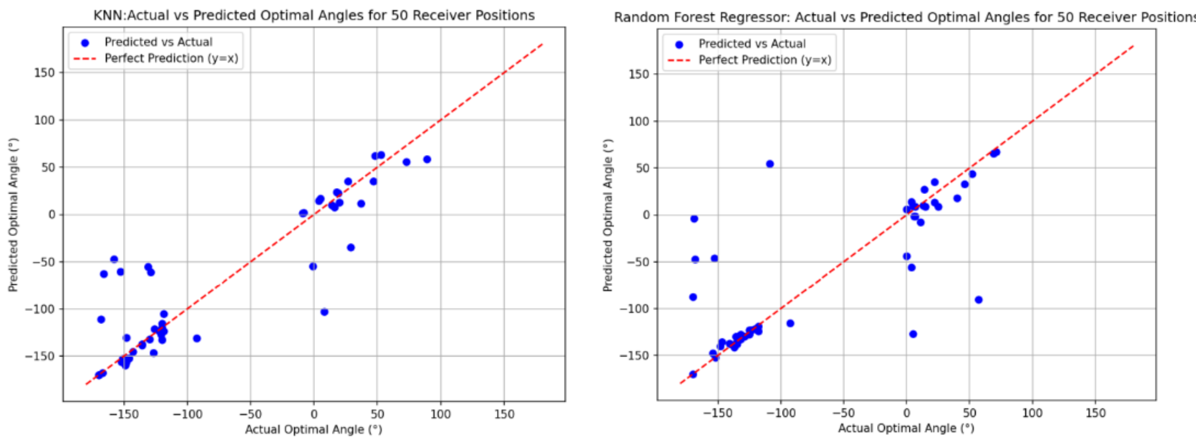


Figure 3-7. Model Evaluation Metrics for N-Nearest Neighbors and Random Forest Regression Models

Model Evaluation Metrics (K-Nearest Neighbors):

MAE (Mean Absolute Error): 21.9°
RMSE (Root Mean Squared Error): 43.4°
R^2 Score: 0.7200

Model Evaluation Metrics (Random Forest Regression):

MAE (Mean Absolute Error): 16.4°
RMSE (Root Mean Squared Error): 40.2°
R^2 Score: 0.8081

Both supervised learning models showed lower performance compared to Deep-Q Network model. The higher MAE and RMSE values indicate that both models made larger prediction errors compared to DQN. The comparison highlights that DQN is a better model for this project as it interacts with the environment, refining its prediction based on the feedback it gets (high reward for optimal angle and low reward for suboptimal ones). In contrast supervised models are limited to learning from fixed, labeled data pairs (input -> output), and cannot interact or adapt dynamically with the environment.

3.3. Software Defined Radio (SDR) Experiments

3.3.1. Introduction

Wireless dead zones, especially in indoor environments, can disrupt communication by weakening or blocking signals. This experiment explores how a simple passive reflector, a copper sheet, can emulate the effects of an RIS to enhance signal coverage. By strategically positioning the sheet between a transmitter and receiver, we aimed to improve signal strength and data throughput.

3.3.2. Procedure

For this experiment, we positioned a transmitter and receiver across a room using a software-defined radio. To simulate a dead zone, foam blocks were placed to obstruct the signal path. A copper sheet was then introduced as a passive RIS-like surface to reflect and enhance the signal. Figure 3-8 shows a picture of this experimental setup within the laboratory.

Figure 3-8 Experimental Setup for SDR Testing of Copper Sheets

The sheet was carefully adjusted at different angles to optimize reflection, with the goal of achieving a measurable improvement in signal gain. Once a noticeable gain was observed on the receiver terminal (a peak around -60 dB for our experiments) the next step was to analyze data transmission performance under different modulation schemes.

Data was transmitted from the USRP while monitoring throughput across two modulation formats: Binary Phase Shift Keying (BPSK) and 16-Quadrature Amplitude Modulation (16-QAM). These two modulation techniques were chosen to observe how a passive RIS-like structure influences both simpler and more complex signal encodings.

3.3.3. Results

The results were then analyzed using constellation plots and EVM diagrams generated by DWSL's DragonRadio image to quantify improvements in signal clarity and reliability. The plots collected from these experiments are shown below in Figure 3-9, Figure 3-10, Figure 3-11 and Figure 3-12.

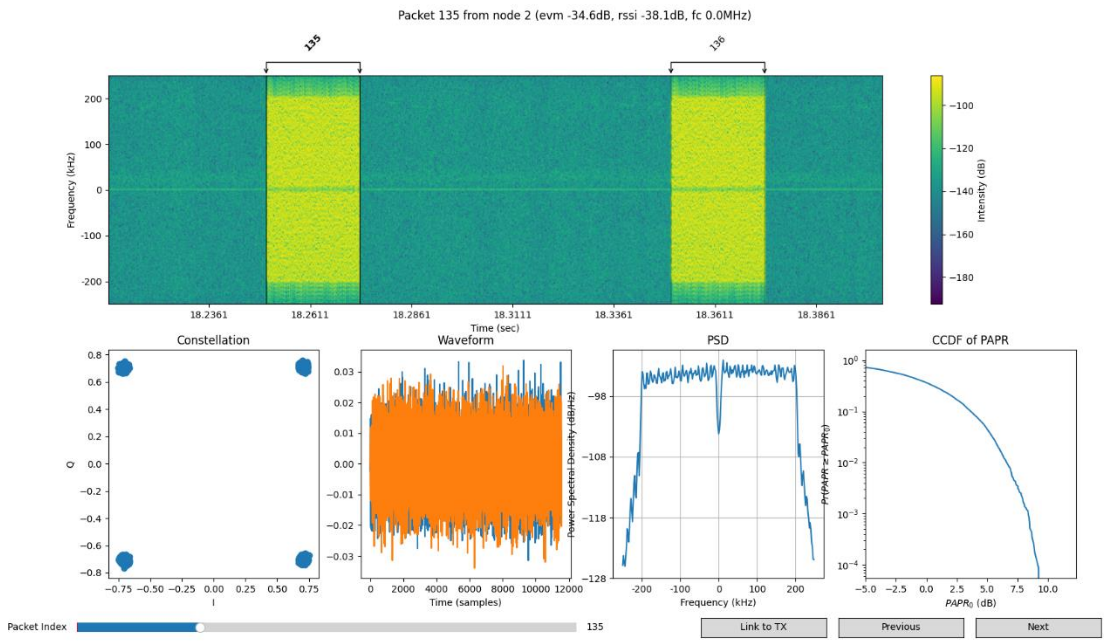


Figure 3-9 BPSK Transmission Metrics Snapshot on the DragonRadio GUI

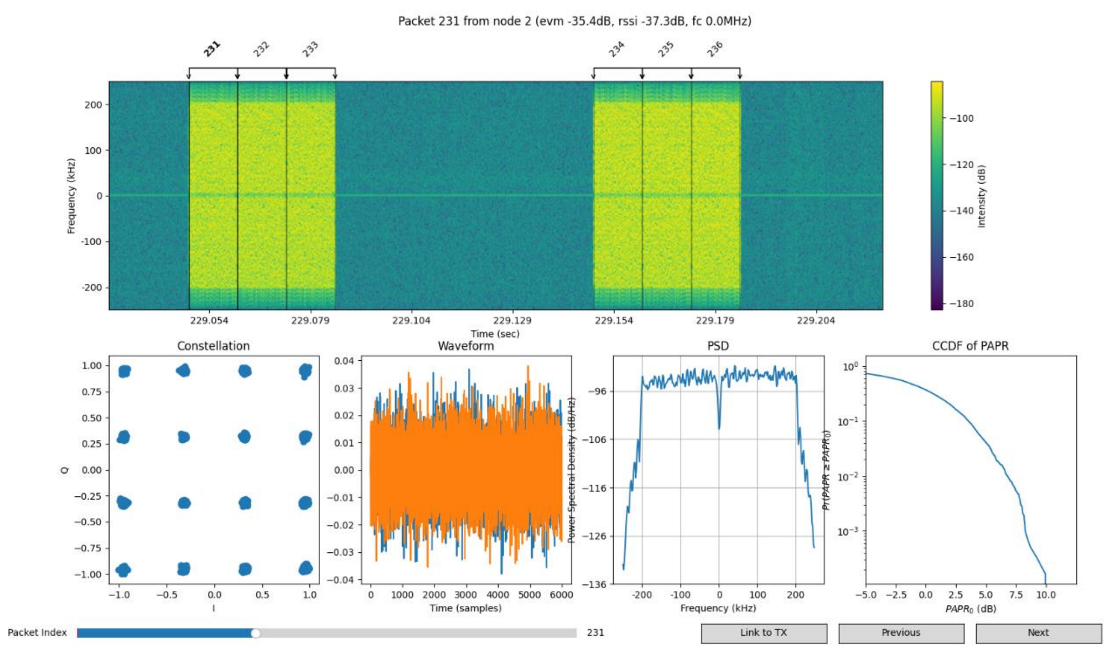


Figure 3-10 QAM16 Transmission Metrics Snapshot on the DragonRadio GUI

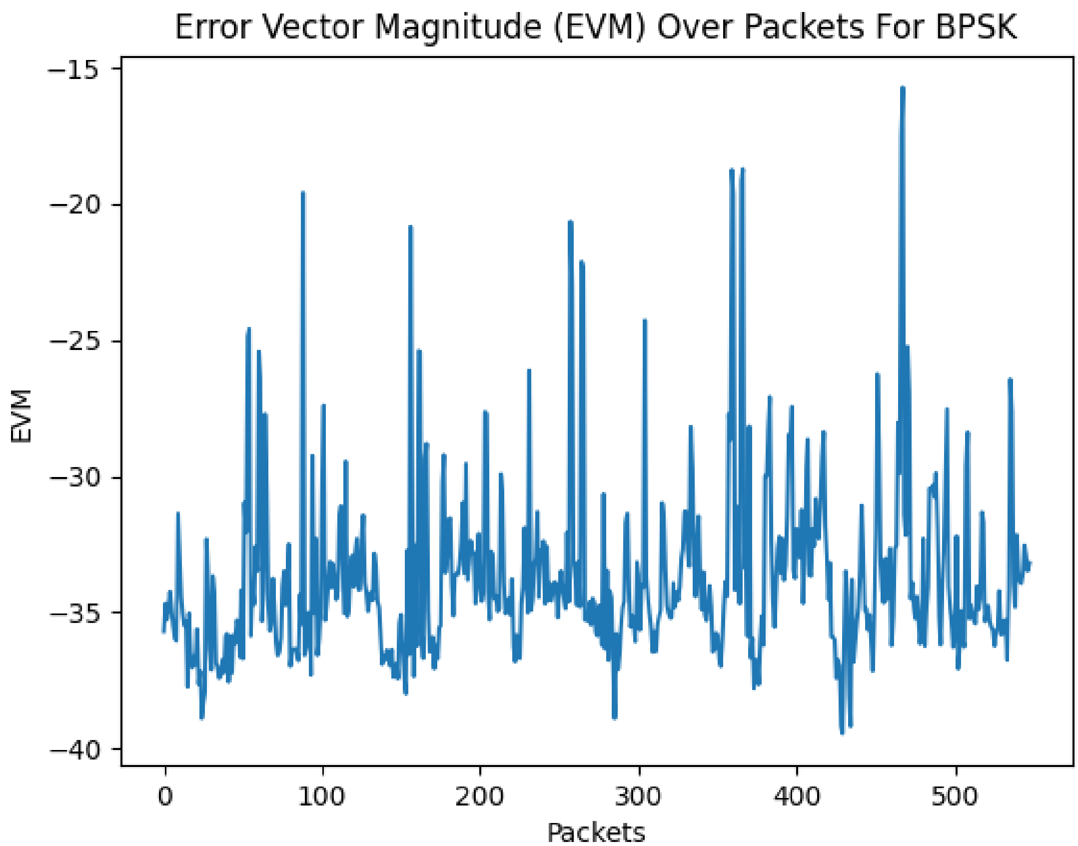


Figure 3-11 Error Vector Magnitude (EVM) Graph for BPSK

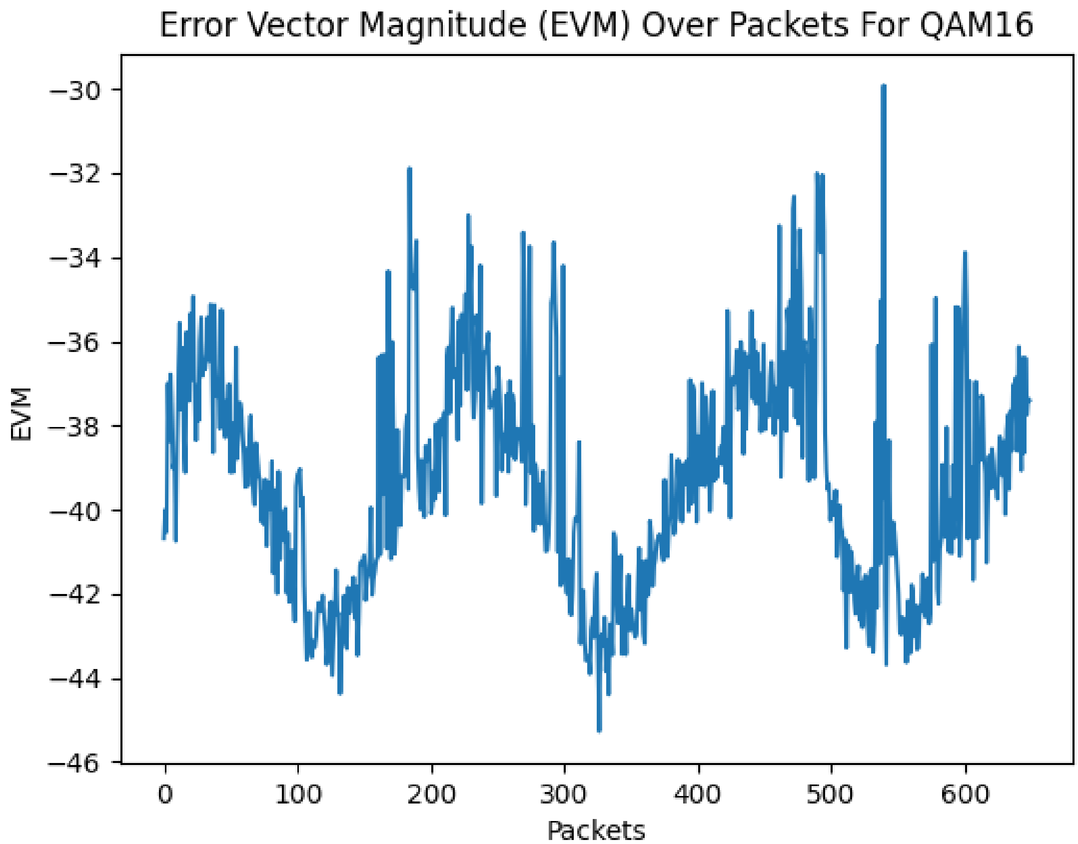


Figure 3-12 Error Vector Magnitude (EVM) Graph for QAM16

3.4. Embedded Design

The RIS controller plays a pivotal role in enabling intelligent reconfiguration of the surface. As previously mentioned, it acts as a middleware between our machine learning algorithm and the antenna elements that are being configured to redirect incident signals. Figure 8-2 in Appendix B demonstrates the full dataflow that results in an optimal configuration of the RIS surface. The controller is responsible for tuning the antenna elements connected to the GPIO pins. The design of this portion of the system prioritizes speed adapting to changes in the channel as close to real-time as possible.

To minimize latency in processing incoming configuration signals, we utilize the onboard direct memory access (DMA) controller to directly configure the GPIO pins connected to the antenna elements. By offloading this task to the DMA controller, we reduce CPU overhead and improve real-time responsiveness. Control signal data is transferred through a buffer from the communication peripheral, inter-integrated circuit (I2C), with the Edge AI device using two DMA controllers. A visual representation of the embedded system handling the surface configuration is shown below in Figure 3-13.

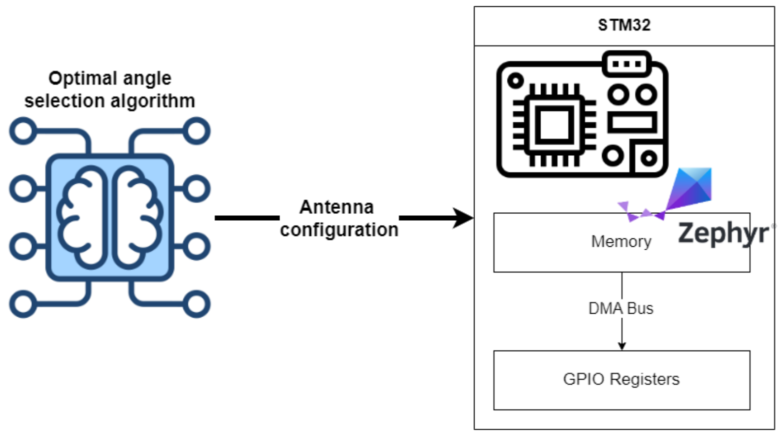


Figure 3-13 Embedded System Control Diagram

4. Recommendations

4.1. Additional Simulations

Wireless InSite simulates the RIS by allowing the designer to input a reflection angle that corresponds with the behavior of the entire board. We have yet to simulate actual configuration of the board considering the individual RF elements of the RIS. We plan to simulate these configurations to prove that we can achieve the reflection angles that we have up until now assumed are possible.

We also plan on simulating more accurate depictions of indoor RIS performance. This would include simulations with a room that resembles the lab that we work in instead of an arbitrary shape. We will also add features that are commonly present in infrastructure such as columns and furniture to evaluate the impact these things have on the indoor performance of an RIS.

4.2. DYSE Integration

While this experiment successfully demonstrated the feasibility of using a simple passive reflector to improve wireless communication, future work will focus on evaluating RIS performance in more dynamic environments. A key next step involves integrating synthetic channel models using DYSE, a machine developed by the Drexel Wireless Systems Lab.

DYSE can take channel coefficients generated from Wireless InSite simulations and feed them into SDR receivers, allowing us to compare real-world passive RIS experiments with deep-Q learning-optimized RIS configurations.

By leveraging DYSE, we aim to verify the authenticity of AI-driven RIS implementations and assess their potential in practical wireless applications. This will provide deeper insights into how RIS technology can be optimized for real-world use, particularly in complex environments where signal enhancement is critical.

4.3. Embedded Design and Machine Learning Integration

The edge AI device should be able to compute the optimal RIS configuration based on the channel estimation supplied to it and send those control signals to the middleware. Our controller, operating the middleware, should update the antenna configuration based on the control signals generated by the edge AI device with as little latency as possible. This enables us to measure the latency of the middleware and control layers and identify areas that can be optimized to achieve reconfiguration as close to real-time as possible.

To achieve this, it requires development of a simple program on the edge TPU development board to communicate the output of the reflection angle optimization algorithm to the surface controller. Once the communication link between the edge TPU device and the middleware controller has been established, we can test the integration of the two devices and measure the time that elapses between our algorithm receiving a channel estimation and the surface being reconfigured.

5. Budget Update

The current budget for this project remains the same as our proposed budget. The budget encompassed the hardware necessary for edge AI computation and control of the surface. Two development boards, a Google Coral for edge AI computing and a STMicroelectronics NUCLEO-F767ZI development board for the middleware layer, were purchased at the beginning of the project and are still the only purchases that are to be accounted for in the budget. Other materials, such as software licensing and access, are provided by the Drexel Wireless Systems lab. Table 8-7 in Appendix D shows our newly proposed budget.

6. Project Management Update

Some adjustments needed to be made to reflect our team's findings within the prototyping and experimentation stage. The updated description of the distribution of roles and responsibilities among team members and our team's Gantt Chart can be found Table 8-5 and Table 8-6 respectively in Appendix C.

7. References

[1] A. A. Puspitasari and B. M. Lee, "A Survey on Reinforcement Learning for Reconfigurable Intelligent Surfaces in Wireless Communications," Sensors (Basel), vol. 23, no. 5, p. 2554, Feb. 2023. doi: 10.3390/s23052554. [PMID: 36904758; PMCID: PMC10007301].

[2] A. Bainbridge, K. Grant, L. Clark and J. Kettle, "Evaluation of the Environmental Impact within Semiconductor Packaging Materials," 2023 24th European Microelectronics and Packaging Conference & Exhibition (EMPC), Cambridge, United Kingdom, 2023, pp. 1-4, doi: 10.23919/EMPC55870.2023.10418417.

[3] B. Yang *et al.*, "Spectrum-Learning-Aided Reconfigurable Intelligent Surfaces for 'Green' 6G Networks," *IEEE network*, vol. 35, no. 6, pp. 20–26, 2021, doi: 10.1109/MNET.110.2100301.

[4] R. F. Ibarra-Hernández, F. R. Castillo-Soria, C. A. Gutiérrez, A. García-Barrientos, L. A. Vásquez-Toledo, and J. A. Del-Puerto-Flores, "Machine Learning Strategies for Reconfigurable Intelligent Surface-Assisted Communication Systems—A Review," Future Internet, vol. 16, no. 5, p. 173, 2024. [Online]. Available: https://doi.org/10.3390/fi16050173

[5] S. Basharat, S. A. Hassan, H. Pervaiz, A. Mahmood, Z. Ding, and M. Gidlund, "Reconfigurable Intelligent Surfaces: Potentials, Applications, and Challenges for 6G Wireless Networks," *IEEE wireless communications*, vol. 28, no. 6, pp. 184–191, 2021, doi: 10.1109/MWC.011.2100016.

[6] STMicroelectronics, "Product longevity program," STMicroelectronics, [Online]. Available: https://www.st.com/content/st_com/en/about/quality-and-reliability/product-longevity.html.

[7] X. Pei *et al.*, "RIS-Aided Wireless Communications: Prototyping, Adaptive Beamforming, and Indoor/Outdoor Field Trials," *IEEE transactions on communications*, vol. 69, no. 12, pp. 8627–8640, 2021, doi: 10.1109/TCOMM.2021.3116151.

[8] Y.-C. Liang *et al.*, "Reconfigurable intelligent surfaces for smart wireless environments: channel estimation, system design and applications in 6G networks," *Science China. Information sciences*, vol. 64, no. 10, pp. 200301-, 2021, doi: 10.1007/s11432-020-3261-5.

8. Appendices

8.1. Appendix A

8.1.1. Machine Learning Decision Matrices

Category	**Concept**
Supervised Learning	Model learns from labeled data that includes both input and output information to predict optimal RIS configurations
Unsupervised Learning	The model identifies patterns in data without labeled input-output pairs. It finds hidden structures in data and is often used in clustering and anomaly detection.
Reinforcement Learning	Model learns through interaction with the environment (trial-and-error method) and receives feedback to optimize RIS configuration for better performance

Table 8-1 Types of ML Models Considered

Criteria	**Supervised Learning**	**Unsupervised Learning**	**Reinforcement Learning**
Data Requirements	Requires large, labeled datasets (input and output) which can be costly and time-consuming to obtain	Learns from unlabeled data but lacks direct supervision	No need for labeled data. Learns from interaction with the environment
Adaptability	Limited; Retraining is required when new data or conditions are introduced	Moderate; Lacks feedback on its actions and struggles with real-time changes	High; continuously Learns and optimizes as the environment evolves making it effective in changing contexts
Training Time	Moderate (depends on dataset size and complexity of model)	High (takes longer to learn meaningful patterns due to lack of supervision)	High (Longer due to exploration and feedback loops)

Table 8-2 Machine Learning Decision Matrix

8.1.2. Embedded System Design Matrix

Criteria	**Microcontroller**	**FPGA**
Power consumption	Low (1 W)	Low-Medium (1-5W)
Workload	Excels at low-complexity control tasks	Excels at parallel computing and intensive tasks
Development time	Uses high-level programming languages (C/C++)	Requires knowledge of low-level programming, hardware-description language (HDL)
Hardware cost	Cost-effective (~ $20/board)	Expensive (~ $80/board)

Table 8-3 Embedded Device Decision Matrix

8.1.3. SDR Decision Matrix

Design Criteria	***Weight***	***SDR (Ranking)***	***SDR (Score)***	***Wi-Fi Card (Ranking)***	***Wi-Fi Card (Score)***
Measurement Capabilities	60%	5	3.0	2	1.2
Versatility	25%	5	1.25	3	0.75
Ease of Use	15%	2	0.3	5	0.75

Total Score	100%		**4.55/5**		**2.7/5**

Table 8-4 Wi-Fi Card vs SDR Decision Matrix

8.2. Appendix B

8.2.1. Proposed System Overview

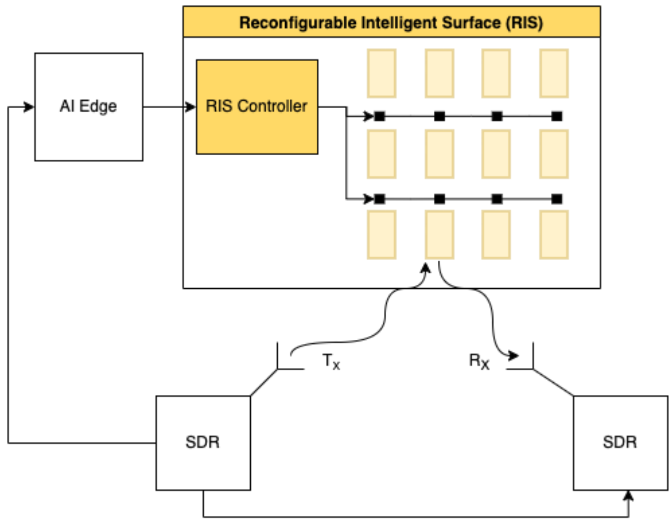


Figure 8-1 Proposed RIS System Overview

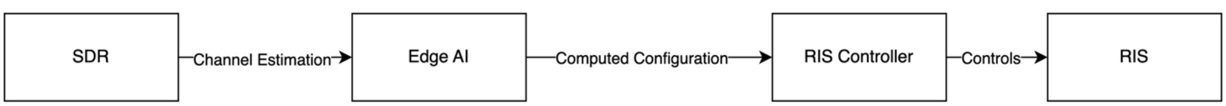


Figure 8-2 Configuration control dataflow

8.3. Appendix C

8.3.1. Organizational Chart

Team Member	Roles	Responsibilities
Brian Cole	• Embedded Programmer • Team Effort Tracker	• Program middleware to facilitate two-way communication between microcontroller and AI edge device • Program RIS matrix control protocol • Ensure a balanced workload between engineers • Lead discussions regarding team goals and expectations
Loukas Athanitis	• RF Simulation and Analysis Engineer • Meeting Organizer	• Simulate RIS models using HFSS/WI • Give channel information to Mari for DSP • Arrange meetings based on advisor and team member availability and establish meeting goals
Mari Takizala	• Digital Signal Engineer • Documentation Manager	• Develop channel estimation algorithms and software-defined radio (SDR) test benches • Observe project deadlines and ensure timely delivery of forms and reports
Surabhi Rajbhandari	• AI/ML Engineer • Design Prototype Manager	• Implement machine learning algorithm for optimizing signal, RIS configuration, beamforming, and channel estimation • Assess team goals and ensure that each engineer is accomplishing individual tasking

Table 8-5. Roles and Responsibilities Organizational Chart

8.3.2. Updated Gantt Chart

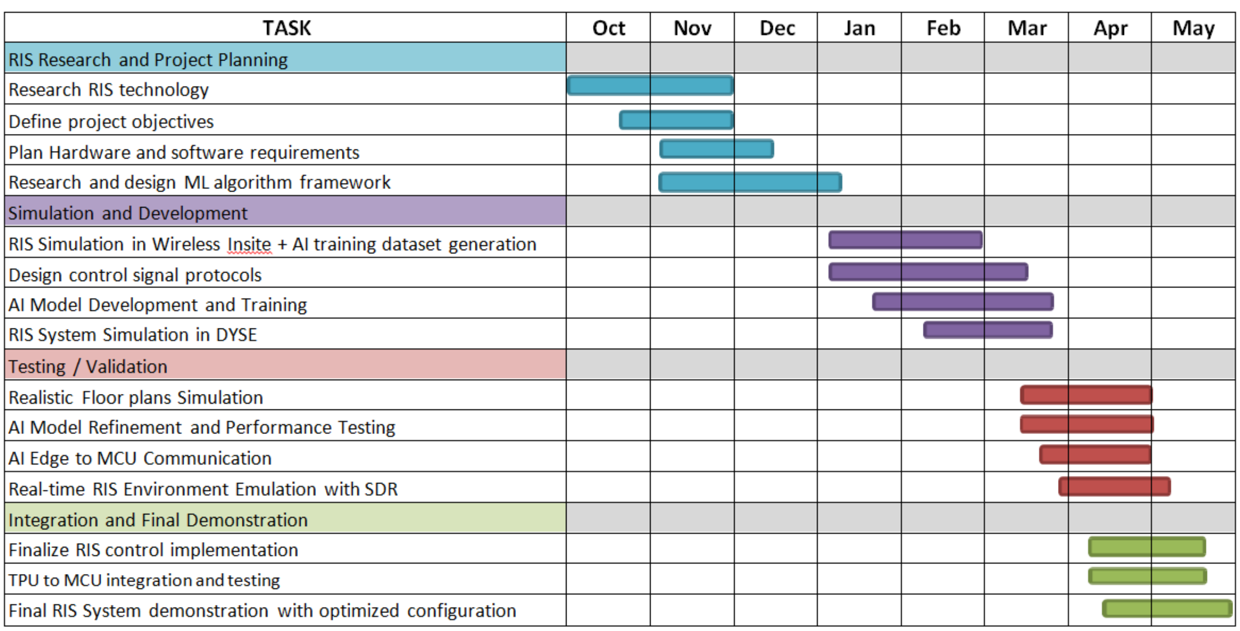


Table 8-6. Updated Gantt Chart

8.4. Appendix D

8.4.1. Updated Budget

System Components	Microcontroller x2	$47.00
	AI Edge Device	$130.00
	Subtotal	**$177.00**

Table 8-7. Updated Budget