

Intelligent Reflecting Surfaces for Wireless Communications: A Machine Learning Approach to Joint Optimization of Beamforming and Phase Vector

Thesis submitted in partial fulfilment of the requirements
for the award of the degree of

Master of Technology

by

Ankit Sharma

(Roll No. 23M1065)

Under the Guidance of

Prof. Prasanna Chaporkar

Co-Guidance of

Prof. Kumar Appaiah



Communication Engineering

Department of Electrical Engineering

INDIAN INSTITUTE OF TECHNOLOGY BOMBAY

Mumbai - 400076, India

June 30, 2025

Thesis Approval

This thesis entitled **Intelligent Reflecting Surfaces for Wireless Communications: A Machine Learning Approach to Joint Optimization of Beamforming and Phase Vector** by **Ankit Sharma**, Roll No. 23M1065, is approved for the degree of **Master of Technology**.

Prof. Sibi Raj B Pillai
(Chairperson & Examiner)

Prof. Kumar Appaiah
(Examiner)

Prof. Prasanna Chaporkar
(Supervisor)

Date:
Place:

Declaration

I declare that this written submission represents my ideas in my own words and where others ideas or words have been included, I have adequately cited and referenced the original sources. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. I understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

.....

Ankit Sharma

Roll No.: 23M1065

Date:

Place:

Abstract

Intelligent reflecting surfaces (IRS) plays a major role in enhancing wireless communication environment, particularly in non-line-of-sight environments. But optimizing IRS phases and beamforming vector jointly is non convex and computational intensive problem. The work begins by analyzing Alternating Optimization (AO) and its computational complexity. In order to decrease the computation time, machine learning (ML) based approaches are explored to predict IRS phase shifts and beamforming vectors from channel gains, aiming to provide a better starting point for these algorithms and accelerate convergence. In single-user scenarios, ML models significantly reduces the algorithmic iterations required by AO. However, for two-user setups, all learning methods perform similar to random initialization.

To further reduce complexity and improve learning feasibility, I adopted the Projected Gradient Method (PGM), which converges faster than AO. I also investigated techniques like phase quantization and IRS element grouping. Quantizing IRS phases into discrete levels learning complexity for ML models but also result in decrease in achievable rate. Grouping IRS elements into square patches also reduced the number of target variables to be predicted, though it resulted in performance loss as patch size increased. I found that combining quantization and element grouping offered a good trade-off between performance and computational complexity for ML models.

We also extended to more complex scenarios: multiple IRSs with a single user and a single IRS with multiple users. In addition, in scenarios with multiple IRSs and multiple users. Our work demonstrates that how joint optimization of phase and beamforming vector can make it less complex for machine learning models.

Contents

| | |
|--|------|
| List of Figures | viii |
| List of Tables | x |
| 1 Introduction | 1 |
| 2 Intelligent Reflective Surfaces | 4 |
| 2.1 Introduction | 4 |
| 2.2 IRS Structure and Control Mechanism | 5 |
| 2.3 IRS-Modified Channel Model | 6 |
| 2.4 Benefits of IRS-Assisted Wireless Communications | 8 |
| 2.4.1 Easy Deployment & Energy Efficiency | 8 |
| 2.4.2 Passive Beamforming and Interference Control | 8 |
| 2.4.3 Capacity and SE/EE Improvements | 9 |
| 2.4.4 Emerging Use Cases | 9 |
| 2.4.5 Comparison with Related Technologies | 10 |
| 2.4.6 Challenges and Research Directions | 10 |
| 2.4.6.1 Channel Estimation | 10 |
| 2.4.6.2 IRS-Tx/Rx Coordination | 11 |
| 2.4.6.3 Optimization Complexity | 11 |
| 2.4.6.4 Realistic Energy Models | 11 |
| 2.4.7 Summary | 11 |
| 3 Machine Learning | 13 |
| 3.1 Introduction | 13 |
| 3.2 Categories of Machine Learning | 14 |
| 3.3 Benefits of Machine Learning in Wireless Communication | 17 |
| 3.4 Key Applications of Machine Learning in Wireless Networks | 20 |
| 3.5 Challenges and Future Research Directions in Machine Learning for Wireless Systems | 23 |

| | |
|--|-----------|
| 4 Intelligent Reflecting Surfaces and Machine Learning | 27 |
| 4.1 Introduction | 27 |
| 4.2 Optimization-Based Joint Design Techniques | 28 |
| 4.3 Machine Learning for IRS Phase Optimization | 29 |
| 5 Problem Statement and Potential Solution | 31 |
| 5.1 Introduction | 31 |
| 5.2 System Model and Notation | 31 |
| 5.3 Complexity Analysis of Baseline Algorithms | 35 |
| 5.4 Learning-Based Initialization for Alternating Optimization | 35 |
| 6 Machine learning based Alternating Optimization: Data Generation and Models | 37 |
| 6.1 Data Generation for Machine Learning Models | 38 |
| 6.1.1 System Setup | 38 |
| 6.1.2 Channel Modeling and Simulation | 38 |
| 6.1.3 Label Generation Using AO Algorithm | 39 |
| 6.1.4 Feature Representation | 39 |
| 6.1.5 Target Representation | 40 |
| 6.1.6 Dataset Structure | 40 |
| 6.2 Machine Learning Models | 41 |
| 6.2.1 Joint Prediction Model | 41 |
| 6.2.2 Separate Prediction Models | 42 |
| 6.2.3 Sequential Prediction Model | 43 |
| 6.3 Results | 44 |
| 6.3.1 Single-User Scenario | 44 |
| 6.3.2 Multi-User Scenario | 48 |
| 6.3.3 Conclusion | 51 |
| 7 Project Gradient Method: Simplification of Data and Extended Use Cases | 52 |
| 7.1 Convergence Speed and Computational Complexity of PGM | 53 |
| 7.1.1 Convergence Speed | 53 |
| 7.1.2 Computational Complexity | 53 |
| 7.2 System Model and Achievable Rate Optimization | 54 |
| 7.2.1 System Architecture | 54 |
| 7.2.1.1 Indirect Link Channels (H_1 and H_2) | 56 |
| 7.2.1.2 Line of Sight Channel (H) | 57 |
| 7.2.2 Achievable Rate Optimization | 57 |
| 7.3 Proposed Projected Gradient Method for 1 User, 1 IRS | 59 |
| 7.3.1 System Model | 59 |
| 7.3.2 Algorithm | 59 |

| | | |
|----------|--|-----------|
| 7.3.3 | Results | 60 |
| 7.3.4 | Effects of Quantization of Phases | 61 |
| 7.3.4.1 | Algorithm Updates | 62 |
| 7.3.4.2 | Results | 62 |
| 7.3.5 | Effect of IRS Element Grouping | 63 |
| 7.3.5.1 | Algorithm Updates | 64 |
| 7.3.5.2 | Results | 64 |
| 7.4 | Proposed Projected Gradient Method for 1 User, 2 IRS | 65 |
| 7.4.1 | Sytem Model | 65 |
| 7.4.2 | Algorithm | 65 |
| 7.4.3 | Results | 67 |
| 7.5 | Proposed Projected Gradient Method for 2 User, 1 IRS | 68 |
| 7.5.1 | Sytem Model | 68 |
| 7.5.2 | Algorithm | 70 |
| 7.5.3 | Results | 73 |
| 7.6 | Proposed Projected Gradient Method for 2 User, 2 IRS | 74 |
| 7.6.1 | Sytem Model | 74 |
| 7.6.2 | Algorithm | 76 |
| 7.6.3 | Results | 80 |
| 8 | Observations and Future work | 81 |
| 8.1 | Observations | 81 |
| 8.2 | Future work | 82 |
| | Bibliography | 84 |

List of Figures

| | | |
|-----|---|----|
| 2.1 | Various use cases of IRS | 5 |
| 2.2 | Schematic of a IRS unit cell with controller. | 6 |
| 2.3 | IRS based communication system of single user. | 7 |
| 5.1 | System Model for alternating optimization | 32 |
| 6.1 | WSR vs. AO Iterations for $K = 1$, $N_t = 2$, $N_{IRS} = 25$ using different initialization strategies. Sequential ML initialization leads to faster convergence. | 45 |
| 6.2 | Box plots for Error percentage of WSR with respect to baseline optimal solution for each iteration for single user scanerio (Phases and Beamforming are predicted using Approach 1). | 45 |
| 6.3 | Box plots for Error percentage of WSR with respect to baseline optimal solution for each iteration for single user scanerio (Phases and Beamforing are predicted using Approach 2). | 46 |
| 6.4 | Box plots for Error percentage of WSR with respect to baseline optimal solution for each iteration for single user scanerio (Phases and Beamforming are predicted using Approach 3). | 46 |
| 6.5 | Box plots for Error percentage of WSR with respect to baseline optimal solution for each iteration for single user scanerio (Phases and Beamforming are predicted using Random Initialization). | 47 |
| 6.6 | WSR vs. AO Iterations for $K = 1$, $N_t = 2$, $N_{IRS} = 25$ using different initialization strategies. Sequential ML initialization leads to faster convergence. | 48 |
| 6.7 | Box plots for Error percentage of WSR with respect to baseline optimal solution for each iteration for single mutli user scanerio(Phases and Beamforming are predicted using Approach 1). | 49 |
| 6.8 | Box plots for Error percentage of WSR with respect to baseline optimal solution for each iteration for single mutli user scanerio (Phases and Beamforming are predicted using Approach 2). | 49 |
| 6.9 | Box plots for Error percentage of WSR with respect to baseline optimal solution for each iteration for single mutli user scanerio (Phases and Beamforming are predicted using Approach 3). | 50 |

| | | |
|------|--|----|
| 6.10 | Box plots for Error percentage of WSR with respect to baseline optimal solution for each iteration for mutli user scanerio (Phases and Beamforming are predicted using Random Initialization). | 50 |
| 7.1 | Aerial view of the considered system, single user, single IRS. | 54 |
| 7.2 | Achivable rate by PGM for single user single IRS | 60 |
| 7.3 | Achievable rate by PGM for quantized phases. | 62 |
| 7.4 | Achivable rate for PGM for grouping IRS elements | 64 |
| 7.5 | Aerial view of the considered system, single user, double IRS. | 65 |
| 7.6 | Achievable rate by PGM for single user, doubler IRS | 67 |
| 7.7 | Aerial view of the considered system, double user, single IRS. | 69 |
| 7.8 | Achievable rate by PGM for double user, single IRS | 73 |
| 7.9 | Aerial view of the considered system, double user, double IRS. | 75 |
| 7.10 | Achievable rate by PGM for double user, doubler IRS. | 80 |

List of Tables

| | | |
|-----|---|----|
| 2.1 | Comparison of IRS, Relay, and Backscatter Technologies | 10 |
| 6.1 | Joint Prediction Model Architecture | 42 |
| 6.2 | Separate Beamforming Predictor Architecture | 42 |
| 6.3 | Separate Phase Predictor Architecture | 43 |
| 6.4 | Phase Predictor Architecture (Using Beamformer Predictor output as input) | 43 |
| 7.1 | Simulation Parameters for Single-User, Single IRS scenario | 60 |
| 7.2 | Simulation Parameters for Single-User, Double IRS scenario | 67 |
| 7.3 | Simulation Parameters for Double-User, Single IRS scenario | 72 |
| 7.4 | Simulation Parameters for Double-User, Double IRS scenario | 79 |

Chapter 1

Introduction

Intelligent Reflecting Surfaces (IRS), also referred to as reconfigurable intelligent surfaces are planar meta surface structures comprising many passive reflecting elements that can adjust the phase and potentially amplitude of incident electromagnetic waves. By intelligently tuning these elements, an IRS enables the dynamic reconfiguration of the wireless propagation environment to improve communication performance. Notably, IRS technology can create favorable indirect paths in scenarios where direct line-of-sight is blocked, thus enhancing coverage and signal quality in non-line-of-sight (NLoS) conditions. Owing to their passive nature, IRSs operate with very low power consumption and add negligible thermal noise, making them an energy-efficient solution for beyond-5G wireless networks. Early studies and applications of IRS depicted in [1] have demonstrated significant gains in system capacity, coverage, and spectral efficiency. For example, improvements have been reported in wireless power transfer, physical layer security, and cognitive radio systems by deploying IRSs to deliberately shape the channel environment.

. The joint design of the active beamforming at the transmitter and the passive beamforming (phase shifts) at the IRS is critical to maximize network throughput. However, this joint optimization problem is non-convex and highly challenging, the transmit beamformer and IRS phase shifts are deeply coupled. The focus of this

research is the maximization of system throughput (sum-rate or achievable rate) in an IRS-assisted wireless network by jointly optimizing the transmitter beamforming vectors and the IRS phase shift configuration. We assume perfect channel state information (CSI) is available for all relevant links (transmitter-to-receiver, transmitter-to-IRS, IRS-to-receiver) to isolate the design problem under ideal conditions. Even with perfect CSI, finding the optimal beamforming and phase strategy is difficult due to the enormous search space and non-convex constraints. Traditional solutions have employed iterative algorithms that alternate between optimizing the transmit beamforming (assuming fixed IRS phases) and optimizing the IRS phases (assuming fixed transmit beamforming). This Alternating Optimization (AO) approach can converge to a locally optimal solution, but it often requires a large number of iterations to reach convergence, especially when the number of IRS elements is large

The computational complexity per iteration is also high because each update may involve solving a sizable optimization sub-problem. The complexity grows rapidly with the number of IRS elements and users, making real-time implementation challenging for large-scale IRS. Recent research [2], [3] has sought more efficient optimization techniques as well as alternative machine learning based approaches to handle the joint design problem.

The PGM algorithm suggested in [2] leverages the gradient of the achievable rate with respect to the phase shifts and projects each gradient update onto the feasible set of phase values. This method was shown to converge faster than the conventional AO approach, achieving the same achievable rate as an AO benchmark but with significantly fewer iterations and reduced computational complexity.

Intuitively, PGM updates all phase elements in a more guided manner using gradient information, rather than the potentially slower alternating approach which updates one set of variables at a time. Despite algorithmic advances like PGM, the complexity of optimizing hundreds or thousands of IRS elements in real time remains a major hurdle. This challenge has motivated exploration of machine learning (ML)

solutions for IRS control. In theory, an ML model (such as a deep neural network) could be trained offline to learn the mapping from environment/channel conditions to good phase shift configurations, thereby bypassing heavy computations during runtime. ML approaches offer the potential for fast inference once trained, and can be robust to uncertainties or incomplete information.

However, purely data-driven methods also face difficulties, including the need for large training datasets covering many channel realizations and generalization to new scenarios. In this work, we aim to use machine learning strengths to provide smart initialization or predictions to reduce the burden on iterative optimization, and using proven optimization algorithms (like AO or PGM) to fine-tune solutions to ensure optimal performance. By doing so, we expect to significantly reduce the iteration count and overall computation required for the joint beamforming and IRS phase optimization problem, enabling a more practical deployment of IRS in dynamic wireless networks.

Chapter 2

Intelligent Reflective Surfaces

2.1 Introduction

In modern time due dense structures and fast varying environment achieving optimal network performance in modern wireless communication systems is a complex challenge. Traditional optimization approaches predominantly focus on transmission-side enhancements such as at base stations (BS) and user equipment (UE), while the propagation environment is largely treated uncontrollable and stochastic. However, signal interactions with the environment including reflection, diffraction, and scattering introduce multi path fading lead to high signal attenuation, ultimately degrading performance in terms of energy efficiency (EE), spectrum efficiency (SE), and reliability.

To overcome these inherent limitations, Intelligent Reflecting Surfaces (IRS) have emerged as a game-changing paradigm. IRSs transform the radio propagation environment from a passive medium into an active, programmable interface, enabling dynamic control over the wireless channel. These surfaces are typically constructed from metasurfaces engineered planar structures comprising sub wavelength sized passive elements whose reflection properties can be tuned in real time.

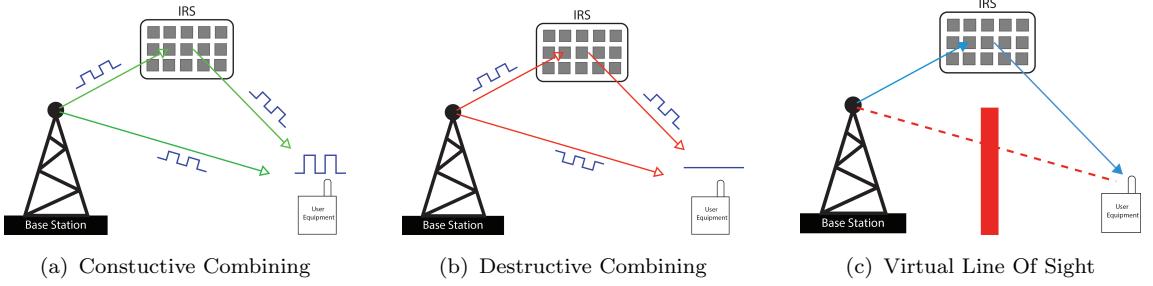


FIGURE 2.1: Various use cases of IRS

Each element can independently adjust the phase (and sometimes amplitude) of incident electromagnetic (EM) waves, allowing the surface to steer reflected signals toward intended directions. This smart manipulation facilitates:

- Constructive combining of desired signals (to enhance power),
- Destructive combining of interference (to suppress unwanted signals),
- Virtual line of sight (LoS) paths in non-line-of-sight (NLoS) environments.

2.2 IRS Structure and Control Mechanism

An IRS is composed of a planar array of passive elements embedded on a reconfigurable metasurface. Each scattering element, typically a metal or dielectric patch, is designed to impose a controllable phase shift on the incident signal. These shifts can be adjusted via some external controller resulting in desired phase shifts. Mostly due to the passive nature of IRS, they are not able to increase the power of incoming signal and can only change its phase. IRSs are typically controlled by a central controller via wired or wireless links. This controller adjusts the phase shifts of each element to suit the communication goal. The IRS itself is passive; it doesn't transmit or decode signals and hence has negligible energy consumption, making it ideal for green communication. These methods allow continuous or quantized tuning of the elements response, enabling:

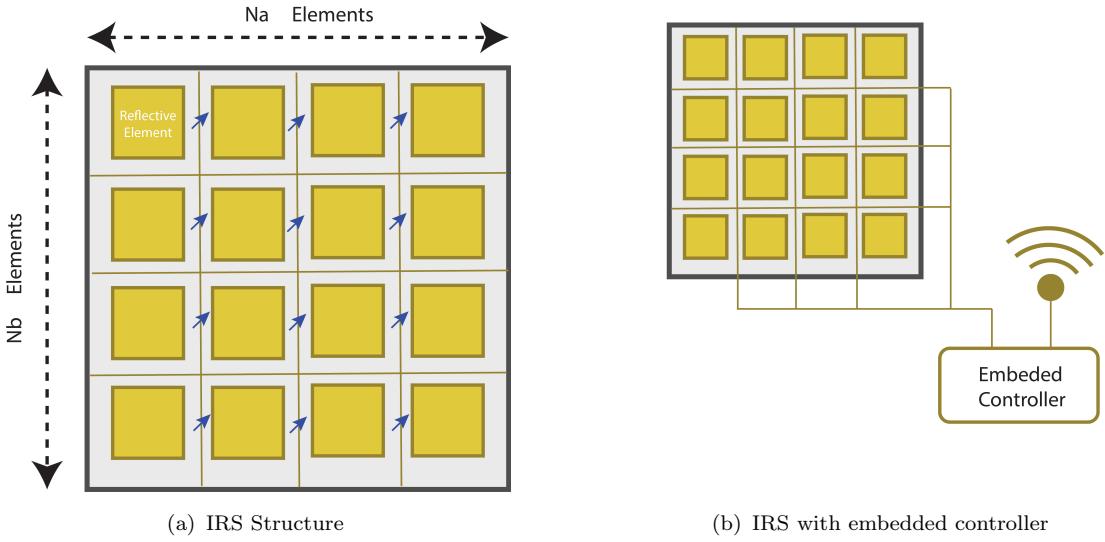


FIGURE 2.2: Schematic of a IRS unit cell with controller.

- Beam steering
- Focusing
- Wavefront shaping

2.3 IRS-Modified Channel Model

Let us consider a single-user MISO (multiple-input, single-output) system with an IRS deployed between a base station and a user.

- Let:
 - $\mathbf{N}_t \in \mathbb{Z}$: Number of base station antennas
 - $\mathbf{N}_r \in \mathbb{Z}$: Number of user antennas
 - $\mathbf{N}_{ris} \in \mathbb{Z}$: Number of IRS elements
 - $\mathbf{H}_1 \in \mathbb{C}^{N_{ris} \times N_t}$: BS-to-IRS channel

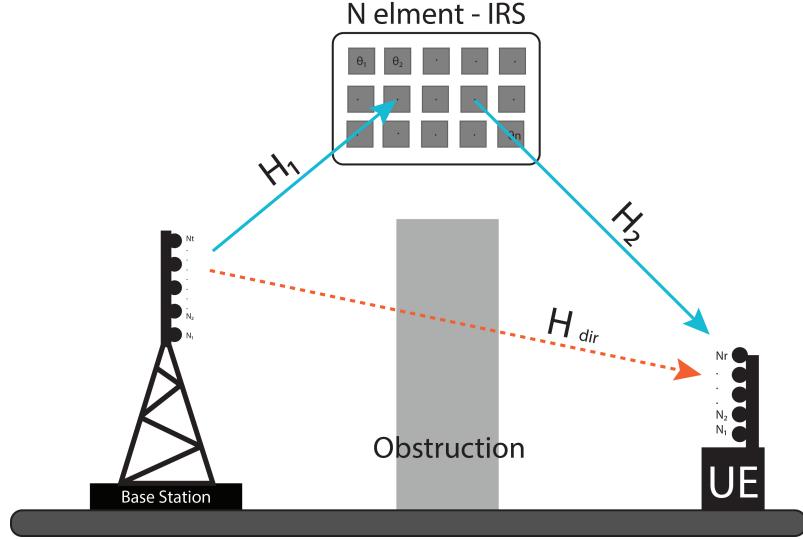


FIGURE 2.3: IRS based communication system of single user.

- $\mathbf{H}_2 \in \mathbb{C}^{N_r \times N_{ris}}$: IRS-to-user channel
- $\boldsymbol{\Theta} = \text{diag}(e^{j\theta_1}, \dots, e^{j\theta N_{ris}})$: IRS diagonal phase-shift matrix
- $\mathbf{H}_{\text{dir}} \in \mathbb{C}^{N_r \times N_t}$: Direct BS-to-user channel

The effective end-to-end channel is:

$$\mathbf{H}_{\text{eff}} = \mathbf{H}_{\text{dir}} + \mathbf{H}_2 \boldsymbol{\Theta} \mathbf{H}_1$$

If the transmitted signal is $\mathbf{X} \in \mathbb{C}^{N_t \times 1}$, and the transmit power is P , then the received signal is:

$$y = \mathbf{H}_{\text{eff}} \mathbf{X} + n$$

where $n \sim \mathcal{CN}(0, \sigma^2)$ is complex Gaussian noise. This model illustrates how IRS modifies the wireless channel without requiring active transmission, enabling passive beamforming for coverage and quality enhancement.

2.4 Benefits of IRS-Assisted Wireless Communications

2.4.1 Easy Deployment & Energy Efficiency

Intelligent Reflecting Surfaces (IRSs) have significant advantages in deployment and energy efficiency, setting them apart from traditional communication technologies. Their passive nature, combined with a lightweight and modular design, makes them incredibly straightforward to install in various environments. Unlike active communication systems, IRSs do not require complex RF chains or power-hungry active signal processing, drastically simplifying their infrastructure. This inherent simplicity also translates into remarkably low energy consumption. In fact, IRSs can often be powered through highly efficient methods like RF harvesting or a minimal back-haul connection, requiring only ultra-low energy to operate. These characteristics collectively make IRSs an appealing solution for future wireless networks, enabling easier and more sustainable deployment of advanced communication capabilities.

2.4.2 Passive Beamforming and Interference Control

Intelligent Reflecting Surfaces (IRSs) offer powerful capabilities in passive beamforming and interference control by intelligently manipulating the wireless environment. Through the optimization of their reflection matrix (Θ), IRSs can precisely shape the propagation of radio frequency (RF) signals. This allows for the enhancement of desired signals by creating constructive interference, effectively focusing the signal towards the intended receiver. Simultaneously, IRSs can suppress unwanted interference by generating destructive interference, effectively nulling out signals from interfering sources. Furthermore, this precise control enables beam splitting, a technique where a single incident signal can be divided and directed to multiple users simultaneously, thereby facilitating efficient multi-user service. These abilities make

IRSSs a game-changer for optimizing signal quality and managing interference in complex wireless networks.

2.4.3 Capacity and SE/EE Improvements

Intelligent Reflecting Surfaces (IRSSs) hold significant potential for dramatically improving the capacity and spectral/energy efficiency (SE/EE) of wireless communication systems. By precisely controlling the phase and amplitude of reflected signals, an IRS can coherently combine the reflected signal with the direct signal at the receiver. This coherent combining leads to a substantial increase in the received Signal-to-Noise Ratio (SNR), which directly translates to higher data rates and improved communication capacity. Furthermore, by creating more efficient propagation paths, IRSSs can effectively reduce the required transmit power from the base station or user device, leading to considerable energy savings. In more complex Multi-User Multiple-Input Multiple-Output (MU-MIMO) setups, the modular nature of IRSSs allows for the flexible allocation of subsets of IRS elements to different users. This capability enables tailored signal steering and interference management for individual users, further optimizing overall system performance and resource utilization.

2.4.4 Emerging Use Cases

The versatile capabilities of Intelligent Reflecting Surfaces (IRSSs) are paving the way for several emerging use cases that promise to revolutionize various aspects of wireless communication. For instance, in Physical Layer Security, IRSSs can play a crucial role by strategically steering “nulls” in the signal propagation. This means they can direct the signal in such a way that it intentionally avoids eavesdroppers, significantly enhancing data confidentiality. In the realm of Wireless Power Transfer (WPT), IRSSs can more effectively direct electromagnetic energy towards receiving

devices, boosting efficiency and range for wirelessly charging gadgets. When it comes to Unmanned Aerial Vehicle (UAV) Networks, the inherent mobility of drones often leads to unstable communication channels. IRSs can compensate for this by dynamically adjusting their reflections, thereby stabilizing the channel and ensuring more reliable communication. Finally, for Mobile Edge Computing (MEC), where computational tasks are offloaded to nearby edge servers, IRSs can improve the efficiency of the uplink communication, enabling faster and more reliable data transfer from user devices to the edge for processing. These diverse applications highlight the transformative potential of IRS technology across a wide spectrum of wireless functionalities.

2.4.5 Comparison with Related Technologies

TABLE 2.1: Comparison of IRS, Relay, and Backscatter Technologies

| Feature | IRS | Relay | Backscatter |
|---------------------|---------------------|------------------|--------------------|
| Power consumption | Passive (ultra-low) | Active (high) | Passive (low) |
| Function | Reflect + beamform | Decode & forward | Reflect + modulate |
| Channel estimation | Indirect | Direct | Direct |
| Hardware complexity | Low | High | Moderate |

2.4.6 Challenges and Research Directions

This section explores the main challenges and research areas for intelligent reflecting surfaces.

2.4.6.1 Channel Estimation

High-dimensional IRS channels lead to training overhead. Efficient estimation and cascaded channel learning methods (e.g., compressed sensing) are areas of active research.

2.4.6.2 IRS-Tx/Rx Coordination

Protocols must support synchronization and joint optimization with BS and UE. Distributed optimization and federated control could be key solutions.

2.4.6.3 Optimization Complexity

IRS beamforming is inherently non-convex due to the discrete/continuous phase constraints. Heuristic, machine learning, and metaheuristic algorithms (e.g., reinforcement learning, PSO, GA) are being investigated.

2.4.6.4 Realistic Energy Models

Studies must include:

- IRS control circuitry energy costs,
- Delay and switching overheads.

2.4.7 Summary

Intelligent Reflecting Surfaces (IRSs) represent an important advancement in the fornext-generation wireless communication. At its core, this innovative technology transforms the communication environment into an actively programmable interface. This fundamental shift is what allows IRSs to offer such substantial advantages.

The capabilities of IRSs are manifold, it can enhance energy efficiency across wireless networks, leading to more sustainable and economical operations. They are also crucial for significantly improving coverage, extending reliable connectivity to areas that might otherwise experience poor signal reception. Furthermore, IRSs excel at suppressing interference, ensuring cleaner and more robust signal transmission,

which is vital for high-quality communication. Beyond these primary benefits, IRSs are key to enabling a variety of novel applications, opening new possibilities for how we interact with wireless technology.

Despite facing certain implementation challenges, such as complex design and control, the far-reaching potential of IRS in 6G and beyond firmly establishes it as a foundational technology. It is set to play a critical role in the evolution of smart and reconfigurable wireless networks, shaping the future of connectivity.

Chapter 3

Machine Learning

3.1 Introduction

Machine Learning (ML) is a field of artificial intelligence (AI) that focuses on enabling computer systems to learn from data without being explicitly programmed. Instead of following a set of pre-defined instructions, ML algorithms are designed to identify patterns, make predictions, and discover insights from vast amounts of information. This learning process allows them to improve their performance over time as they are exposed to more data. The core idea is to train a model using examples, much like how humans learn from experience. Once trained, this model can then be used to make decisions or predictions on new, unseen data. This capability makes ML a powerful tool for solving complex problems where traditional rule-based programming is either too difficult or impossible.

In recent years, Machine Learning (ML) has emerged as a transformative paradigm in engineering and technology, particularly in the domain of wireless communications. Traditional wireless communication systems rely heavily on model-based design, where channel models, noise statistics, and transmission strategies are assumed to be known or estimable. However, these models often fail to capture the complex

and dynamic nature of real-world environments, especially in the context of 5G and beyond, where devices, mobility, and service requirements vary rapidly. This limitation becomes even more apparent with the increasing demand for higher data rates, lower latency, and massive connectivity, which traditional methods struggle to efficiently manage.

Machine learning provides a powerful alternative by learning patterns and decision policies directly from data. Instead of relying on analytical models, ML algorithms adapt and generalize from observations, making them particularly suitable for wireless systems characterized by non-linearity, high dimensionality, and temporal variability. These advantages enable ML to assist or even replace traditional signal processing and resource allocation algorithms across various layers of the wireless communication stack. For instance, ML can be used to improve tasks like channel estimation, predicting how wireless signals will travel, or interference management, helping devices avoid stepping on each others signals. It can also optimize how resources like bandwidth and power are shared among many users, making the network more efficient. This data-driven approach allows wireless networks to become more intelligent, flexible, and capable of handling the unpredictable nature of modern communication demands.

3.2 Categories of Machine Learning

Machine Learning is a powerful field that allows computers to learn from data without being explicitly programmed. It is broadly divided into three main ways of learning, each suited for different kinds of problems.

1. Supervised Learning

In supervised learning, we train a computer model using data that already has the answers. For example it is like teaching a child with flashcards: you show them a picture of a dog (input) and tell them it is a “dog” (output). The goal is for the model to learn from these examples so it can guess the right answer when it sees new, un-seen data.

Example Applications in Wireless Communication:

- **Channel State Estimation:** Predicting how wireless signals will travel through the air, which helps in better communication.
- **Signal Classification:** Identifying different types of wireless signals.
- **Modulation Recognition:** Figuring out how a signal has been encoded.
- **User Mobility Prediction:** Guessing where a mobile user might move next, which helps networks prepare.

This method is very common when you have a lot of historical data with clear labels, and you want to make predictions or categorize new information.

2. Unsupervised Learning

Unsupervised learning is different because the data does not come with answers. Instead, the model tries to find hidden patterns or structures within the data on its own. For example, it is like giving a child a pile of toys and asking them to sort them into groups without telling them what the groups should be (e.g., all blocks together, all cars together).

Example Applications in Wireless Communication:

- **User Clustering for Beamforming:** Grouping users with similar needs so that network antennas can focus signals more effectively.
- **Traffic Pattern Detection:** Finding typical ways that network data flows.
- **Anomaly Detection in Networks:** Spotting unusual activities that might indicate a problem or a security threat.

Unsupervised learning is very useful when you have a lot of data but do not know what you are looking for, or when it is too difficult to label the data by hand.

3. Reinforcement Learning (RL)

Reinforcement learning is like teaching by trial and error. An “agent” (our computer program) learns to make decisions by trying things out in an environment. It gets “rewards” for good actions and “penalties” for bad ones. Over time, it learns the best actions to take to get the most rewards. For example, teaching a robot to walk, it falls down a lot at first (penalty), but when it takes a step without falling (reward), it learns what to do.

Example Applications in Wireless Communication:

- **Dynamic Spectrum Access:** Deciding which radio frequencies to use at any given moment to avoid interference and maximize efficiency.
- **Power Control:** Adjusting the strength of signals to save energy while maintaining good communication quality.
- **Intelligent Handoff Decisions:** Smartly deciding when a mobile device should switch from one base station to another as a user moves.

- **IRS Phase Shift Optimization:** (This relates to the previous chapter on Intelligent Reflecting Surfaces, or IRS). Here, RL can help optimize how these special surfaces reflect signals to improve wireless links.

RL is particularly powerful for problems where decisions need to be made in a changing environment, and where the outcome of those decisions is not immediately obvious. It is about learning through experience, much like humans do.

3.3 Benefits of Machine Learning in Wireless Communication

The application of Machine Learning (ML) techniques in wireless systems yields numerous advantages, significantly transforming how these networks operate and evolve. MLs ability to learn from data and make intelligent decisions is particularly well-suited for the dynamic and complex nature of wireless communication.

- **Adaptability** ML algorithms can adapt to changing environments, such as varying channel conditions, user densities, and interference levels, without manual tuning. This inherent adaptability is crucial for maintaining optimal performance in real-world wireless scenarios where conditions are rarely static. For instance, an ML-powered system can dynamically adjust its transmission power or modulation scheme in response to real-time changes in signal-to-noise ratio, ensuring robust and efficient data delivery. This contrasts sharply with traditional, static approaches that often struggle to cope with such fluctuations.
- **Model-Free Optimization** ML can optimize system parameters in complex scenarios where no closed-form or tractable model exists, such as joint optimization of beamforming and resource allocation. This capability is invaluable

in situations where the underlying physics or interactions are too intricate to be described by simple mathematical equations. For example, in massive MIMO (Multiple-Input Multiple-Output) systems, ML can learn optimal beamforming patterns directly from observed data, leading to superior spectral efficiency and reduced interference even without a precise channel model. This data-driven optimization opens up new possibilities for performance enhancements that were previously unattainable.

- **Data-Driven Inference** Real-time data streams from sensors, user equipment, and network logs can be used to train models that enable intelligent decision-making. This continuous influx of data allows ML models to constantly refine their understanding of the network state and predict future trends. For example, by analyzing historical traffic patterns and current user behavior, ML can predict network congestion hotspots and proactively reallocate resources to prevent service degradation. This predictive capability moves wireless networks from reactive problem-solving to proactive optimization.
- **Automation and Scalability** ML enables autonomous network control, reducing human intervention and allowing large-scale deployment with minimal overhead. The ability of ML algorithms to learn and operate independently significantly reduces the operational expenditure associated with managing complex wireless networks. For instance, self-organizing networks (SON) powered by ML can automatically configure new base stations, optimize network parameters, and resolve faults without requiring manual input. This automation is critical for scaling up future wireless networks, such as 5G and beyond, which will feature unprecedented numbers of connected devices and diverse service requirements. Furthermore, ML can handle the immense volume of data generated by these large-scale deployments, extracting valuable insights that would be impossible for human operators to process.

- **Enhanced Security** ML can significantly bolster the security of wireless communication systems. By analyzing network traffic patterns and identifying anomalous behavior, ML algorithms can detect and mitigate various cyber threats, including denial-of-service (DoS) attacks, impersonation, and data breaches. For example, an ML model trained on legitimate network behavior can quickly flag unusual connection attempts or data transmissions as potential security breaches, enabling rapid response and protection. This proactive threat detection is crucial in an era of increasingly sophisticated cyberattacks.
- **Improved Resource Management** ML algorithms excel at optimizing the allocation of scarce wireless resources, such as spectrum, power, and computational resources. By predicting future demand and current network conditions, ML can make intelligent decisions about how to distribute these resources to maximize throughput, minimize latency, and ensure fair access for all users. For instance, in dynamic spectrum access, ML can identify underutilized frequency bands and intelligently allocate them to users, leading to more efficient spectrum utilization and reduced interference. This intelligent resource management is key to unlocking the full potential of next-generation wireless technologies.
- **Energy Efficiency** Wireless networks are significant consumers of energy. ML can play a crucial role in optimizing energy consumption across the network. By analyzing traffic load, user distribution, and environmental factors, ML algorithms can intelligently manage the power states of network components, such as base stations and user equipment. For example, an ML-driven system can put underutilized base stations into a low-power sleep mode during off-peak hours, or adjust the transmission power of individual devices to the minimum required for reliable communication. This leads to substantial energy savings, reducing operational costs and contributing to a greener communication infrastructure.

- **Personalized User Experience** ML can enable a more personalized and seamless experience for wireless users. By understanding individual user behavior, preferences, and service requirements, ML algorithms can tailor network services accordingly. For instance, an ML-powered network can prioritize bandwidth for a user streaming high-definition video while dynamically adjusting settings for another user engaging in a low-latency online game. This level of personalization extends to proactive troubleshooting, where ML can predict potential issues based on user-specific data and address them before they impact the users experience.

These benefits collectively highlight the transformative potential of integrating Machine Learning into wireless communication systems, paving the way for more efficient, reliable, secure, and intelligent networks.

3.4 Key Applications of Machine Learning in Wireless Networks

Machine Learning (ML) is being increasingly applied across various facets of wireless networks, bringing about significant improvements in efficiency, performance, and intelligence. Here are some of the key areas where ML is making a substantial impact:

- **Channel Estimation and Prediction** Accurate channel state information (CSI) is fundamental for efficient wireless communication. Traditional methods often rely on sending frequent pilot signals, which consume valuable network resources. ML models, particularly those based on deep learning, can learn complex temporal and spatial correlations within wireless channels. This allows them to effectively estimate channel conditions with reduced pilot overhead and even predict future channel states. For example, a deep neural

network can be trained on a large dataset of past channel measurements to predict how the channel will evolve in the next few milliseconds, enabling proactive adjustment of transmission parameters and improving overall system throughput. This leads to more efficient use of bandwidth and reduced latency.

- **Resource Allocation and Scheduling** Efficient management of wireless resources, such as spectrum, power, and time slots, is critical for maximizing network capacity and ensuring quality of service (QoS). ML algorithms, especially reinforcement learning (RL), are highly effective in optimizing user scheduling, power allocation, and spectrum sharing in real-time. RL agents can learn optimal policies by interacting with the wireless environment, observing the consequences of their actions, and receiving rewards for desired outcomes. For instance, an RL agent can learn to dynamically allocate power to different users based on their channel conditions and service requirements, ensuring fair access and maximizing network throughput. This dynamic and intelligent resource management outperforms static or rule-based approaches, especially in highly dynamic environments.
- **Intelligent Beamforming and Antenna Selection** Beamforming, which directs wireless signals towards specific users, and antenna selection, which chooses the best set of antennas for transmission, are crucial for improving signal strength and reducing interference. ML can significantly enhance these techniques. Using past user mobility and location data, supervised learning models can predict the optimal beam direction or the best antenna subset to use for a particular user at any given time. This allows for more precise and adaptive beamforming, leading to stronger signals at the receiver and reduced interference for other users. For example, a convolutional neural network (CNN) can process channel state information and user location data to output optimal beamforming weights, enhancing spectral efficiency and energy efficiency in complex multi-antenna systems like massive MIMO.

- **Network Traffic Forecasting** Predicting future network traffic load is essential for proactive network management, allowing operators to prevent congestion and optimize resource provisioning. Time-series models, such as Recurrent Neural Networks (RNNs) and particularly Long Short-Term Memory (LSTM) networks, are well-suited for this task. By analyzing historical traffic patterns, these models can accurately predict future traffic surges and dips. This enables dynamic backhaul provisioning, where network capacity can be adjusted in anticipation of demand, and proactive congestion control mechanisms can be activated before performance degrades. For instance, if an LSTM model predicts a significant increase in video streaming traffic in a certain area, the network can pre-allocate additional bandwidth to that region, ensuring a seamless user experience.
- **Security and Anomaly Detection** Wireless networks are vulnerable to various security threats, including unauthorized access, jamming, and data manipulation. ML plays a vital role in enhancing network security by enabling real-time anomaly detection. Unsupervised learning methods are particularly useful here, as they can learn the “normal” behavior of network parameters (e.g., signal strength, packet arrival rates, device activity patterns). Any significant deviation from this learned normal behavior is flagged as an anomaly, potentially indicating malicious users or unusual activities. For example, an autoencoder can be trained on typical network traffic data; if an incoming data packet deviates significantly from the reconstructed output, it could indicate a malicious injection. This proactive detection helps in mitigating security breaches and maintaining the integrity and availability of wireless services.
- **Interference Management** Interference is a major challenge in wireless networks, significantly impacting signal quality and network capacity. ML algorithms can learn complex interference patterns and develop intelligent strategies to mitigate their effects. By analyzing data on active users, channel conditions, and power levels, ML models can predict interference levels and suggest

optimal power control or frequency reuse schemes. This proactive interference management can lead to a cleaner wireless environment, enabling higher data rates and more reliable connections, especially in dense urban areas with many co-existing wireless devices.

- **Device-to-Device (D2D) Communication Optimization** D2D communication allows devices to communicate directly with each other without routing through a base station, offering benefits like increased throughput, reduced latency, and improved energy efficiency. ML can optimize D2D communication by determining when and how devices should engage in direct communication. ML models can assess factors such as distance, channel quality, and traffic load to decide whether D2D communication is beneficial and to optimize power levels and resource allocation for these direct links. This intelligent orchestration of D2D communication maximizes its benefits while minimizing potential interference with traditional cellular communications.

These diverse applications demonstrate how Machine Learning is transforming wireless networks from static, rule-based systems into intelligent, adaptive, and self-optimizing infrastructures, poised to meet the demands of future communication technologies.

3.5 Challenges and Future Research Directions in Machine Learning for Wireless Systems

While the integration of Machine Learning (ML) into wireless communication systems offers transformative potential, it also introduces a unique set of challenges that require significant research and development. Addressing these challenges is crucial for realizing the full benefits of ML in this domain and for its widespread adoption.

- **Data Quality and Availability** Obtaining labeled and representative datasets for training ML models is a significant hurdle in many real-time wireless environments. Wireless channels are inherently dynamic, and collecting exhaustive data that captures all possible scenarios (e.g., varying user densities, interference patterns, environmental conditions) is challenging and resource-intensive. Furthermore, data labeling, especially for complex tasks like anomaly detection or channel prediction, often requires expert knowledge and can be prohibitively expensive. Future research needs to focus on advanced data collection techniques, synthetic data generation, and methods for learning from limited or unlabeled data, such as semi-supervised learning and active learning, to overcome this constraint.
- **Generalization and Transferability** ML models trained in one specific wireless setting may not generalize well across different network topologies, frequency bands, device types, or geographical locations. This lack of generalization means that models often need to be retrained or finely tuned for each new deployment, which can be impractical and costly. Research into transfer learning, federated learning, and domain adaptation techniques is essential to enable ML models to learn from diverse data sources and adapt efficiently to new, unseen environments without extensive re-training. The goal is to develop “universal” ML models that can perform robustly across a wide range of wireless scenarios.
- **Model Complexity vs. Latency** Deep learning models, while offering high accuracy for complex tasks, often come with significant computational overhead and memory requirements. This can lead to unacceptable computational delays (latency) for real-time applications in wireless networks, such as dynamic resource allocation or fast beamforming, where decisions need to be made in milliseconds. There is a critical need for research into developing lightweight ML models, model compression techniques (e.g., quantization,

pruning), and efficient hardware architectures (e.g., specialized AI accelerators) that can execute complex ML inference with low latency and power consumption suitable for edge devices and base stations in wireless networks.

- **Integration with Legacy Systems** Seamlessly embedding ML modules into existing, often proprietary, communication infrastructures without causing disruptions is a non-trivial challenge. Wireless networks are built upon decades of standardized protocols and hardware, and integrating new ML components requires careful design to ensure compatibility, interoperability, and backward compatibility. Future research should explore modular architectures, standardized ML interfaces for wireless components, and software-defined networking (SDN) principles to facilitate the smooth and flexible deployment of ML functionalities within heterogeneous wireless ecosystems. This includes developing frameworks that allow ML models to interact with existing control planes and data planes effectively.
- **Trust, Explainability, and Security** The “black-box” nature of many sophisticated ML models, particularly deep neural networks, raises significant concerns around interpretability, robustness, and susceptibility to adversarial attacks. In critical infrastructure like wireless communication, understanding *why* an ML model makes a particular decision is vital for debugging, compliance, and ensuring reliability. Furthermore, ML models can be vulnerable to carefully crafted adversarial examples that can fool them into making incorrect decisions, posing serious security risks. Future research must focus on developing explainable AI (XAI) techniques for wireless applications, robust ML algorithms that are resilient to adversarial attacks, and verifiable ML methods to build trust in ML-driven wireless systems. Ensuring the privacy of training data in collaborative learning scenarios (e.g., federated learning) is also a crucial aspect of security and trust.

- **Energy Consumption of ML Processing** While ML can optimize energy efficiency in wireless networks, the computational demands of training and even inferencing complex ML models themselves can consume significant energy. This is particularly relevant for edge devices with limited power budgets. Research is needed to develop energy-efficient ML algorithms and hardware specifically tailored for wireless applications. This includes exploring techniques like “green AI”, which focuses on reducing the carbon footprint of AI models, and specialized neural network architectures designed for low-power operation at the network edge.
- **Standardization and Regulation** For widespread adoption, the integration of ML into wireless systems will require new standards and regulatory frameworks. This includes defining common interfaces for ML components, establishing performance metrics for ML-driven wireless functions, and addressing regulatory aspects related to data privacy, security, and the autonomous operation of ML systems in licensed spectrum. Collaboration between industry, academia, and regulatory bodies will be crucial to establish clear guidelines and foster innovation while ensuring safety and fairness.

Addressing these challenges through dedicated research efforts will pave the way for a new era of intelligent, efficient, and secure wireless communication, unlocking the full potential of Machine Learning in future networks.

Chapter 4

Intelligent Reflecting Surfaces and Machine Learning

4.1 Introduction

The rapid advancement of Intelligent Reflecting Surfaces (IRS) has marked a paradigm shift in the design and optimization of wireless communication systems. As a revolutionary enabling technology for 6G and beyond, IRS introduces a new layer of control over the propagation environment by intelligently reconfiguring wireless channels through passive beamforming. This chapter explores the core research directions that have emerged in the IRS field, with a focus on optimization-based joint design techniques and machine learning-based IRS control strategies. These two areas together provide a solid foundation for building efficient, adaptive, and scalable IRS-aided wireless systems.

4.2 Optimization-Based Joint Design Techniques

One of the earliest and most influential works in the optimization-based domain was presented by [4]). They analyzed a multi-user MISO downlink system assisted by a single IRS. The authors formulated the weighted sum-rate (WSR) maximization problem, jointly optimizing the transmit beamforming at the access point (AP) and the phase shifts at the IRS. This problem is inherently non-convex due to the coupling between AP beamforming and IRS phase control. To solve this, they proposed an alternating optimization (AO) approach which we will discuss in subsequent chapter:

1. Fix the IRS phase configuration and optimize the beamforming vectors at the AP (often convex and solvable via water-filling or fractional programming).
2. Fix the beamforming and update the IRS phase shifts, which typically requires solving a non-convex problem using successive convex approximation (SCA) or semidefinite relaxation (SDR).

Despite its iterative nature, the algorithm converges to a stationary point of the original WSR problem. [4] further enhanced efficiency by using closed-form expressions (via fractional programming) for the beamforming subproblem and a stochastic SCA method for phase updates. However, this method becomes computationally intensive with large IRS sizes: the per-iteration complexity of IRS phase updates scales approximately as $\mathcal{O}(K^2 N_{\text{IRS}}^2)$, where K is the number of users.

Expanding upon the single-IRS paradigm, [5] investigated systems equipped with two cooperating IRSs. Their study demonstrated the performance benefits of dual reflections, especially when IRSs are jointly optimized. The double-IRS architecture allows multiple signal paths (e.g., AP → IRS1 → IRS2 → User), increasing both the channel rank and the array gain. Joint optimization (referred to as cooperative passive beamforming) further amplified spectral efficiency and system coverage.

This dual-IRS insight highlights an essential trade-off: although performance improves with more IRS elements or surfaces, the optimization complexity rises sharply, necessitating efficient algorithms or new methods to estimate the IRS phase vector.

4.3 Machine Learning for IRS Phase Optimization

Machine learning (ML) has recently emerged as a powerful tool to tackle the challenges of real-time, adaptive control in IRS-assisted systems. Traditional optimization algorithms, though effective, are often too slow for fast-varying wireless environments. In contrast, ML models can infer optimal or near-optimal IRS phase configurations almost instantaneously once trained.

Supervised Learning and Hybrid Approaches

A promising direction is optimization-assisted ML, where the ML model is trained using labelled data generated by conventional optimization solvers. For example, a deep neural network (DNN) can be trained to predict the IRS phase vector from the channel state information (CSI) or geometric features such as angles of arrival and departure.

Deep Reinforcement Learning (DRL)

Another line of research utilizes deep reinforcement learning (DRL), where an IRS controller acts as an agent interacting with the environment. It learns the optimal phase configuration by maximizing a reward function (e.g., throughput) over time. DRL is attractive for scenarios without full CSI or with time-varying environments. However, it suffers from the increase in dimensionality for because as IRS elements

increases the dimension of search space also increase making exhaustive exploration infeasible. Despite this challenge, DRL remains promising in:

- Coverage improvement in dynamic environments,
- Security enhancement through adaptive beamforming,
- Power minimization under practical constraints.

Successful DRL applications often rely on action space reduction, network pruning, or state aggregation to remain tractable.

Chapter 5

Problem Statement and Potential Solution

5.1 Introduction

In this chapter, we describe the system model for our IRS-aided wireless communication scenario and analyze the data and problem structure that underlie our approach. This includes defining the notation, formulating the optimization problem for joint beamforming and phase shifts, and examining the computational complexity of existing algorithms. We also discuss how data (such as channel realizations and optimization outcomes) is generated and used for training the machine learning models later on

5.2 System Model and Notation

We consider a downlink communication system consisting of a multi-antenna base station or access point (AP), K users, and one intelligent reflecting surfaces (IRSs) deployed to assist the communication. Each user is assumed to have N_r receive

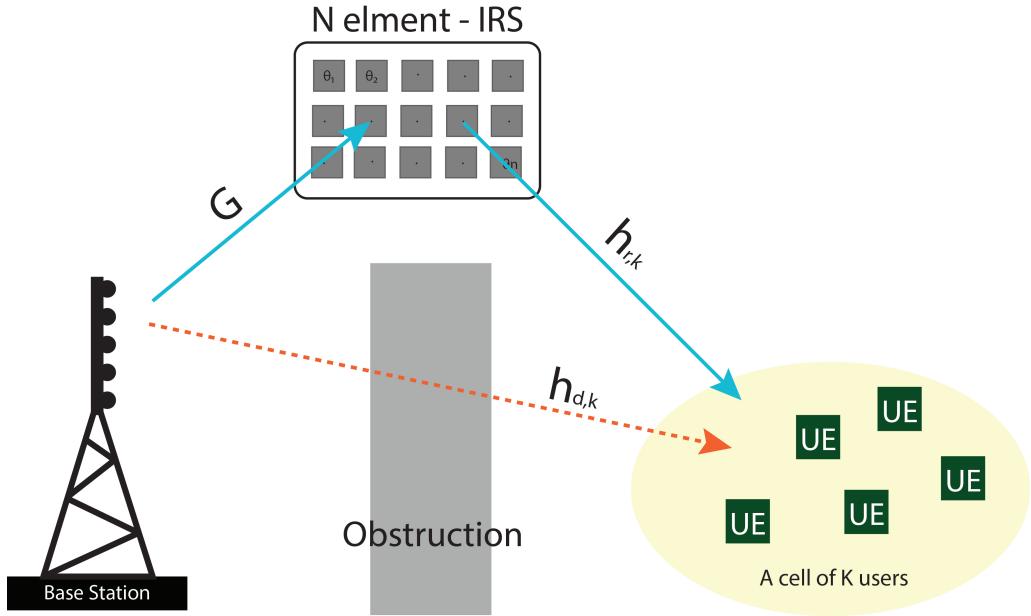


FIGURE 5.1: System Model for alternating optimization

antennas (in many of our simulations $N_r = 1$ for simplicity, i.e., single-antenna users, but the model can accommodate $N_r > 1$ for MIMO users or receivers). The AP is equipped with N_t transmit antennas. The IRS has N_{IRS} passive reflecting elements. Key notation is summarized as follows (similar to the definitions in our short report summary):

- N_{IRS} : Number of IRS elements on a given surface.
- N_t : Number of transmit antennas at the base station (AP).
- N_r : Number of receive antennas per user.
- K : Number of users in the network.

For channels, we denote: - \mathbf{G} as the channel matrix from the AP to the IRS. This has dimension $N_{\text{IRS}} \times N_t$ (assuming each IRS element has a single effective antenna and receives the combined signal from N_t transmit antennas). - $\mathbf{h}_t(r, k)$ as the channel vector from the IRS to user k . If user k has N_r antennas, $\mathbf{h}_t(r, k)$ can be viewed as an $N_r \times 1$ vector. This captures the effect of each IRS elements reflection on the

signal received at user k). - \mathbf{G} matrix, or for single-antenna user $N_r = 1$ it is an $N_{\text{IRS}} \mathbf{h}_d(r, k)$ as the direct channel from the AP to user k (of dimension $N_r \times N_t$ for general MIMO, or a vector of length N_t for single-antenna user).

These channels can be modeled with line-of-sight (LoS) components and/or fading. In our simulations, we typically assume Rician fading: the AP-IRS channel \mathbf{h}_{AP} and IRS-user channel $\mathbf{h}_G(r, k)$ may have a strong LoS component (since IRSs are often strategically placed to have clear links with the AP and the coverage area). The direct channel \mathbf{h}_d could be NLoS or blocked (which is exactly when the IRS provides the most benefit). We also assume **far-field** propagation for simplicity, meaning the distances are such that planar wavefronts and consistent phase shifts across the IRS elements can be assumed (no spherical wavefront variations across the surface). This is valid when the IRS and users are sufficiently far apart relative to the IRS size.

Signal Model:

The AP transmits signals to K users. Let $\mathbf{w}_k \in \mathbb{C}^{N_t}$ be the beamforming vector for user k (applied to the information symbol for user k). The IRS applies a phase shift θ_n on each element $n \in \{1, \dots, N_{\text{IRS}}\}$, or equivalently by a diagonal Φ . We can represent the IRS phase shift configuration by a vector $\theta = [\theta_1, \theta_2, \dots, \theta_{N_{\text{IRS}}}]$ reflection matrix $\Phi = \text{diag}(e^{j\theta_1}, e^{j\theta_2}, \dots, e^{j\theta_{N_{\text{IRS}}}})$, where $e^{j\theta_n}$ is the complex reflection coefficient of element n . In an ideal continuous-phase IRS, $\theta_n \in [0, 2\pi)$ can take any value; in a quantized IRS, θ_n is restricted to a discrete set (e.g., $\{0, \pi\}$ for 1-bit, etc.).

The received signal at user k is then the superposition of the direct signal from the AP and the reflected signal from the IRS. Mathematically, for user k :

$$y_k = \mathbf{h}_{d,k}^H \mathbf{x} + \mathbf{h}_{r,k}^H \Phi \mathbf{G} \mathbf{x} + z_k,$$

where $\mathbf{x} = \sum_{i=1}^K \mathbf{w}_i s_i$ is the transmitted signal (superposition of K users data symbols s_i precoded by \mathbf{w}_i), and z_k is noise at user k . We assume s_i are normalized $E[|s_i|^2] = 1$ and without loss of generality $E[s_i s_j^*] = 0$ for $i \neq j$. The noise z_k is Gaussian with variance σ^2 . The base station has a transmit power constraint, e.g. $\sum_{i=1}^K |\mathbf{w}_i|^2 \leq P$.

The instantaneous **rate** for user k (if treating other users signals as interference) can be written as:

$$R_k = \log_2 \left(1 + \frac{|\mathbf{h}_{d,k} \sum_i \mathbf{w}_i + \mathbf{h}_{r,k} \Phi \mathbf{G} \sum_i \mathbf{w}_i|^2}{\sum_{j \neq k} |\mathbf{h}_{d,k} \mathbf{w}_j + \mathbf{h}_{r,k} \Phi \mathbf{G} \mathbf{w}_j|^2 + \sigma^2} \right).$$

Optimization Problem:

The central problem is to find a set of $\{\mathbf{w}_k\}$ and θ to maximize a performance objective, typically the sum-rate $\sum_{k=1}^K R_k$. In a simpler form, one can consider maximizing the *weighted sum-rate* $\sum_k \alpha_k R_k$ for some weights α_k , or in a single-user case maximize the throughput R_1 directly. This is subject to the power constraint at the transmitter and the unit-modulus constraints $|\exp(j\theta_n)| = 1$ for all IRS elements. Mathematically:

$$\begin{aligned} & \max_{\{\mathbf{w}_k\}, \theta} \quad \sum_{k=1}^K \alpha_k R_k(\{\mathbf{w}_k\}, \theta) \\ & \text{subject to} \quad \sum_{k=1}^K \|\mathbf{w}_k\|^2 \leq P_{\max}, \\ & \quad \theta_n \in [0, 2\pi) \text{ for } n = 1, \dots, N_{\text{IRS}}. \end{aligned}$$

This problem is non-convex due to the coupling of \mathbf{w}_k and θ inside R_k . Even if we ignore interference (as in a single-user case), maximizing the received SNR by jointly designing beamformer and phases is non-trivial. A brute-force search over

θ (especially if continuous) is impossible for large N_r , and even if θ is discrete, the search space is exponential in N_{IRS} .

5.3 Complexity Analysis of Baseline Algorithms

The AO algorithm is the baselines for solving the above joint optimization. We summarize their complexity characteristics here to motivate the need for further improvement. In AO, we alternate between optimizing \mathbf{w}_k (usually via solving a convex problem like a weighted MMSE or water-filling solution if θ is fixed) and optimizing θ (often via coordinate descent or sequential optimization of each phase, or using an approximate method since even optimizing phases for fixed \mathbf{w}_k is non-convex). The complexity of the beamforming optimization step is on the order of $O(KN_t^2)$ if using direct matrix inversion or water-filling for K data streams (this comes from solving linear equations of size N_t or performing singular value decompositions, etc.). The complexity of the phase optimization step can be much higher, often $O(K^2N_{\text{IRS}}^2)$ for naive methods. More sophisticated algorithms like in [4] reduce this by using closed-form updates or iterative scalar optimization per element. AO algorithm as presented in [4] has computational complexity $\mathcal{O}(I_0(2KNirsNt+KNt^2+K^2Nirs^2))$ and relies on random phase initialization. But due SCA or fractional programming step, this complexity becomes prohibitive as N_{IRS} grows large (say hundreds or thousands), the complexity scales on the order of N_{IRS}^2 inside the loop.

5.4 Learning-Based Initialization for Alternating Optimization

The Alternating Optimization (AO) algorithm, as discussed in [4], is widely used for joint beamforming and IRS phase optimization. However, its computational

complexity, given by

$$\mathcal{O}(I_0(2KN_{\text{IRS}}N_t + KN_t^2 + K^2N_{\text{IRS}}^2)),$$

where I_0 is the number of outer iterations, makes it computationally intensive particularly for large-scale IRS systems. Furthermore, the algorithm traditionally relies on random initialization of the IRS phase vector, which can lead to slow convergence or suboptimal solutions.

To address these limitations, learning-based methods have been proposed to provide better initializations for AO. The core idea is to use a neural network to predict a good starting point for the beamforming and IRS phase vectors, based on the observed channel gains. This helps in reducing the number of required AO iterations by starting closer to a high-performing solution.

Three machine learning strategies are examined:

1. **Joint Prediction:** A single model predicts both the beamforming vector and IRS phase vector simultaneously.
2. **Separate Prediction:** Two distinct models are trained independently one for the beamforming vector and another for the IRS phase vector.
3. **Sequential Prediction:** The beamforming vector is predicted first, followed by the IRS phase vector using a second model that conditions on the beamforming output.

In summary, while machine learning provides promising avenues for accelerating AO algorithms in IRS-aided systems, its effectiveness is highly dependent on the system configuration, particularly the number of users and the dimensionality of the optimization problem. In further chapter we will discuss about the data generation and potential these machine learning models can provide to AO algorithms.

Chapter 6

Machine learning based Alternating Optimization: Data Generation and Models

To train effective machine learning (ML) models for predicting IRS phase shifts and beamforming vectors, a high-quality dataset is crucial. In this work, we generate training data by simulating wireless scenarios with IRS-aided communication systems and solving the corresponding optimization problems using the Alternating Optimization (AO) algorithm. The generated dataset includes both input features (channel state information) and output labels (optimal or near-optimal beamforming and IRS phase vectors). Then we will used this tagged data to train different deep learning models. The predicted values would be used as starting point for the AO algorithm in order to decrease the outer iterations required to reach at local optimal solution.

6.1 Data Generation for Machine Learning Models

6.1.1 System Setup

The following settings were used in all simulations:

- The Access Point (AP) is located at coordinates $(0, m, 0, m)$.
- K users are uniformly and randomly distributed within a circular area of radius $10, m$ centered at $(200, m, 30, m)$.
- User priority is inversely proportional to their distance from the AP.
- Perfect Channel State Information (CSI) is assumed.
- Each result is averaged over 100 random test channel realizations.
- For the benchmark BCD algorithm, we used 100 outer iterations, while our learning-based approach used 30 outer iterations.
- Two simulation scenarios were considered:
 - Scenario 1: $N_{ris} = 25$ (IRS elements), $K = 1$ (user), $Nt = 2$ (AP antennas)
 - Scenario 2: $N_{ris} = 50$, $K = 2$, $Nt = 4$

6.1.2 Channel Modeling and Simulation

To model the wireless environment, we simulate both large-scale and small-scale fading effects for each link:

- **Large-scale fading:** Path loss is computed based on the Euclidean distance with a standard attenuation exponent.
- **Small-scale fading:** Rician fading is used to model the channels involving the IRS to capture line-of-sight (LoS) components, while Rayleigh fading may be used for direct AP-user links.

The channels generated for each realization include:

- $\mathbf{h}_{d,k}$: Direct channel from AP to user k
- \mathbf{G} : Channel from AP to IRS
- $\mathbf{h}_{r,k}$: Channel from IRS to user k

To ensure practical relevance, scenarios are selected such that the IRS provides meaningful gains, typically in non-line-of-sight (NLoS) or obstructed cases between the AP and users.

6.1.3 Label Generation Using AO Algorithm

For each simulated channel realization, we use the AO algorithm to approximately solve the joint beamforming and phase optimization problem. The solutions serve as ground truth targets for ML models. Although these methods do not guarantee global optimality, they provide high-quality labels suitable for supervised learning.

6.1.4 Feature Representation

The input features to the ML model must comprehensively represent the wireless environment:

- We use the full channel matrices $\mathbf{h}_{d,k}$, $\mathbf{h}_{r,k}$, and \mathbf{G} .

- The real and imaginary parts of each channel element are extracted and concatenated to form the feature vector.
- In high-dimensional settings, further dimensionality reduction may be considered (e.g., effective channel summaries or dominant path extraction), but for moderate values of N_{IRS} , full channel representations are feasible.

6.1.5 Target Representation

The target variables (labels) for the regression/classification tasks are constructed as follows:

- The beamforming vector \mathbf{w} and phase vector $\boldsymbol{\theta}$ obtained from AO are split into real and imaginary parts.
- These components are concatenated to form the target vector for each sample.

6.1.6 Dataset Structure

Dataset consist of 30,000 sample points. Each data sample in the dataset consists of:

- **Input (features):** Real and imaginary components of channel matrices for all links involved.
- **Output (labels):** Real and imaginary components of the optimal beamforming and IRS phase vectors.

6.2 Machine Learning Models

This is an overview of the machine learning models developed for this project with detailed model architectures, diagrams, and additional evaluations.

We explored three types of neural network models to predict the transmit beamforming vectors and IRS phase shifts from channel state information. All models were implemented in Python using a deep learning framework (such as TensorFlow or PyTorch) and use mean square error (MSE) as loss function. The learning task was approached in different ways depending on the strategy used, which are discussed below.

6.2.1 Joint Prediction Model

In this approach, a single neural network is designed to jointly predict both the beamforming vector \mathbf{w} and the IRS phase vector $\boldsymbol{\theta}$. This model directly maps the input channel state (i.e., real and imaginary parts of \mathbf{h}_d , \mathbf{G} , and \mathbf{h}_r) to the output beamforming and phase configuration. Model architecture is present in table 6.1.

Input and Output Format

- **Input:** Flattened and concatenated channel vectors/matrices with real and imaginary components.
- **Output:**
 - N_t -dimensional beamforming vector (continuous).
 - N_{IRS} -dimensional phase vector (continuous or quantized into Q classes).

TABLE 6.1: Joint Prediction Model Architecture

| Attribute | 1 User | 2 Users |
|----------------------|-------------------------------------|---|
| Input size | 154 | 620 |
| Output size | 54 | 116 |
| No. of hidden layers | 7 | 8 |
| Neurons per layer | 314, 314, 256, 128, 100, 100, 64 | 1024, 1024, 512, 512, 256, 150, 128, 128 |
| Activation functions | ReLU | ReLU |
| Optimizer | Adam | Adam |
| Dropout rate | 2e-3 | 1e-3 |
| L2 regularization | 1e-10 | 0 |
| Test MSE | 0.1369 | 0.1957 |
| Validation MSE | 0.1410 | 0.2483 |

TABLE 6.2: Separate Beamforming Predictor Architecture

| Attribute | 1 User | 2 Users |
|----------------------|-------------------------------|------------------------------------|
| Input size | 154 | 620 |
| Output size | 4 | 16 |
| No. of hidden layers | 6 | 7 |
| Neurons per layer | 314, 256, 128, 100, 70, 64 | 512, 314, 256, 128, 100, 70, 64 |
| Activation functions | ReLU | ReLU |
| Optimizer | Adam | Adam |
| Dropout rate | 1e-3 | 1e-3 |
| L2 regularization | 1e-5 | 1e-5 |
| Test MSE | 0.0057 | 0.0951 |
| Validation MSE | 0.0061 | 0.1089 |

6.2.2 Separate Prediction Models

In this strategy, two separate neural networks are developed:

1. **Beamforming Predictor:** Maps channel features to the beamforming vector.
2. **Phase Predictor:** Maps channel features to IRS phase vector.

This separation allows more flexibility in tuning hyperparameter and architectures specific to each subtask. The model architecture is show in table 6.2 and 6.3 for beamforming predictor and phase predictor respectively.

TABLE 6.3: Separate Phase Predictor Architecture

| Attribute | 1 User | 2 Users |
|----------------------|------------------------|-------------------------------------|
| Input size | 154 | 620 |
| Output size | 50 | 100 |
| No. of hidden layers | 5 | 7 |
| Neurons per layer | 256, 128, 100, 100, 64 | 1024, 1024, 512, 512, 256, 128, 128 |
| Activation functions | ReLU | ReLU |
| Optimizer | Adam | Adam |
| Dropout rate | 1e-3 | 1e-3 |
| L2 regularization | 1e-6 | 10e-8 |
| Test MSE | 0.1777 | 0.1862 |
| Validation MSE | 0.1820 | 0.2600 |

TABLE 6.4: Phase Predictor Architecture (Using Beamformer Predictor output as input)

| Attribute | 1 User | 2 Users |
|----------------------|------------------------|---------------------------|
| Input size | 158 | 636 |
| Output size | 50 | 100 |
| No. of hidden layers | 5 | 5 |
| Neurons per layer | 314, 256, 128, 100, 64 | 2048, 1024, 512, 512, 256 |
| Activation functions | ReLU | ReLU |
| Optimizer | Adam | Adam |
| Dropout rate | 1e-3 | 1e-3 |
| L2 regularization | 1e-5 | 10e-8 |
| Test MSE | 0.1574 | 0.1446 |
| Validation MSE | 0.1591 | 0.2001 |

6.2.3 Sequential Prediction Model

The sequential model cascades two neural networks: the first predicts the beamforming vector, and the second uses both the channel input and the predicted beamformer to estimate the IRS phase configuration. This structure allows learning dependencies between beamforming and phase selection. The model architecture is show in table 6.3 and 6.4 for beamforming predictor and phase predictor respectively.

6.3 Results

6.3.1 Single-User Scenario

We first examine a baseline single-user MISO system with one IRS to evaluate the effect of IRS optimization in the absence of inter-user interference. The simulation parameters for this case are: $K = 1$ user, $N_t = 2$ transmit antennas at the AP, $N_r = 1$ receive antenna at the user, and $N_{\text{IRS}} = 25$ IRS elements. The IRS is strategically positioned to enhance the NLoS path between the AP and the user.

The performance metric considered is the weighted sum-rate (WSR), which reduces to the single users achievable rate in this scenario. Figure 6.10 shows the WSR versus the number of AO iterations under different initialization schemes:

- **Random Initialization:** IRS phase angles are initialized randomly.
- **Joint ML Prediction:** A neural network jointly predicts the transmit beam-forming vector \mathbf{w} and IRS phase vector $\boldsymbol{\theta}$.
- **Separate ML Prediction:** The network predicts \mathbf{w} and $\boldsymbol{\theta}$ independently.
- **Sequential ML Prediction:** Beamforming is predicted first, followed by IRS phase configuration.

Results show that all ML-based initializations yield a significantly higher starting rate compared to random initialization. Among them, the sequential prediction strategy provides the best initialization, allowing the AO algorithm to converge in fewer iterations. This indicates that ML predictions provide a better starting point to algorithm which is much closer to the local optimal solution.

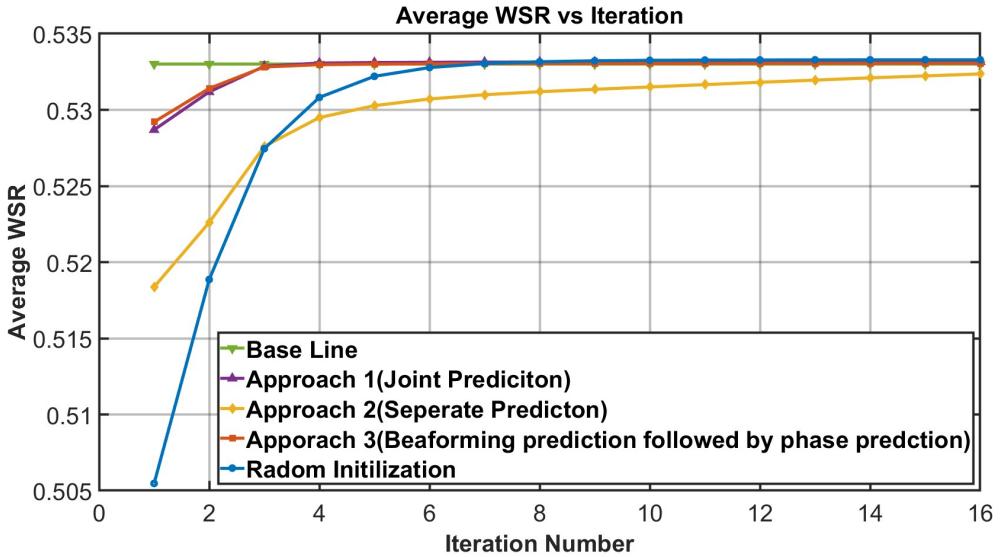


FIGURE 6.1: WSR vs. AO Iterations for $K = 1$, $N_t = 2$, $N_{IRS} = 25$ using different initialization strategies. Sequential ML initialization leads to faster convergence.

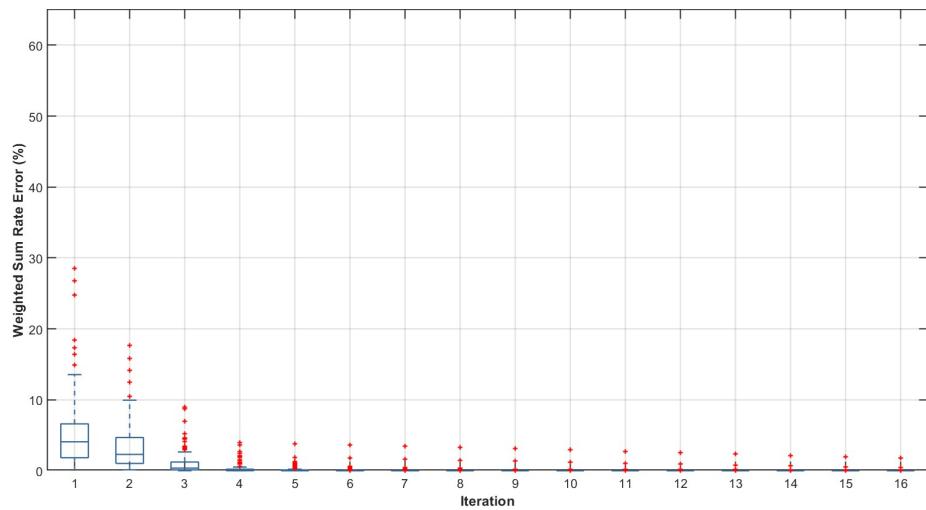


FIGURE 6.2: Box plots for Error percentage of WSR with respect to baseline optimal solution for each iteration for single user scenario (Phases and Beamforming are predicted using Approach 1).

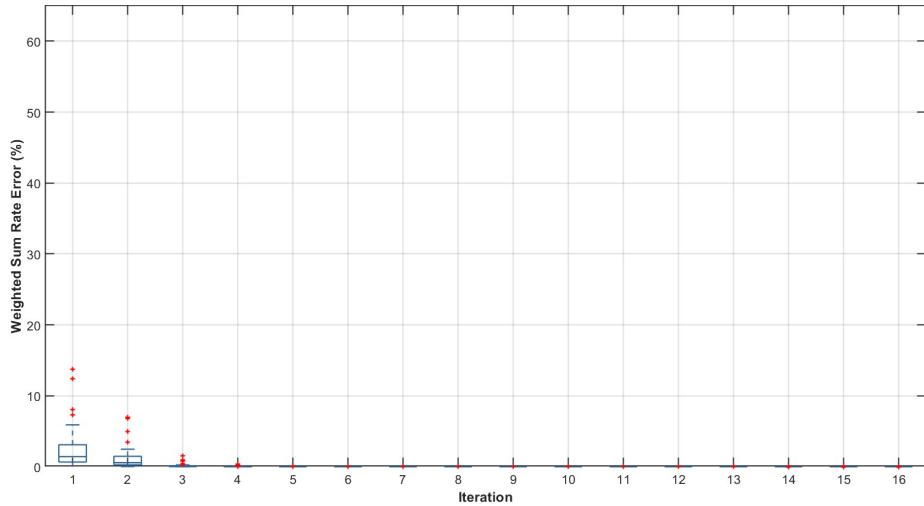


FIGURE 6.3: Box plots for Error percentage of WSR with respect to baseline optimal solution for each iteration for single user scanerio (Phases and Beamforming are predicted using Approach 2).

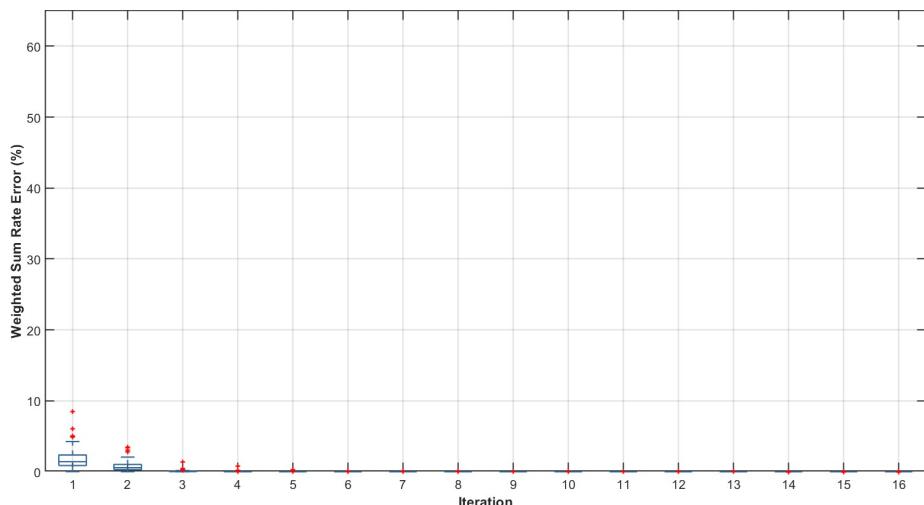


FIGURE 6.4: Box plots for Error percentage of WSR with respect to baseline optimal solution for each iteration for single user scanerio (Phases and Beamforming are predicted using Approach 3).

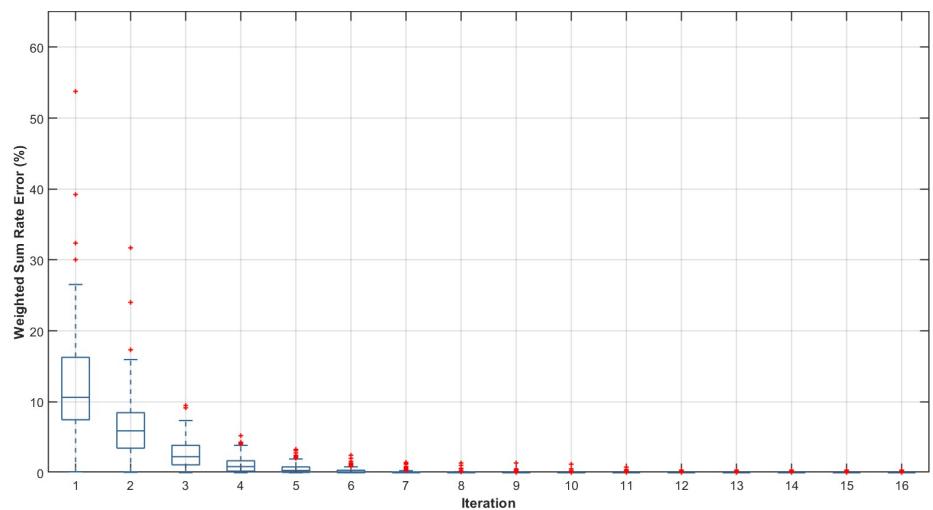


FIGURE 6.5: Box plots for Error percentage of WSR with respect to baseline optimal solution for each iteration for single user scanerio (Phases and Beamforming are predicted using Random Initialization).

6.3.2 Multi-User Scenario

We now analyze the performance in a two-user system ($K = 2$), where $N_t = 4$ antennas and $N_{\text{IRS}} = 50$ elements are used. This setup introduces inter-user interference, making the optimization more complex and the mapping from channel state to optimal configuration harder to learn.

As shown in Figure 6.6, ML-based initialization strategies provided improvement over random initialization in terms of convergence speed or final WSR. All methods eventually converged to similar performance, but separate prediction approach is able to get little more WSR as compare to other approaches. But if we compare it to the single user case, there is decrease in performance, the likely reason is that the interference environment creates a highly non-linear and coupled optimization problem, which is difficult to generalize from training data alone.

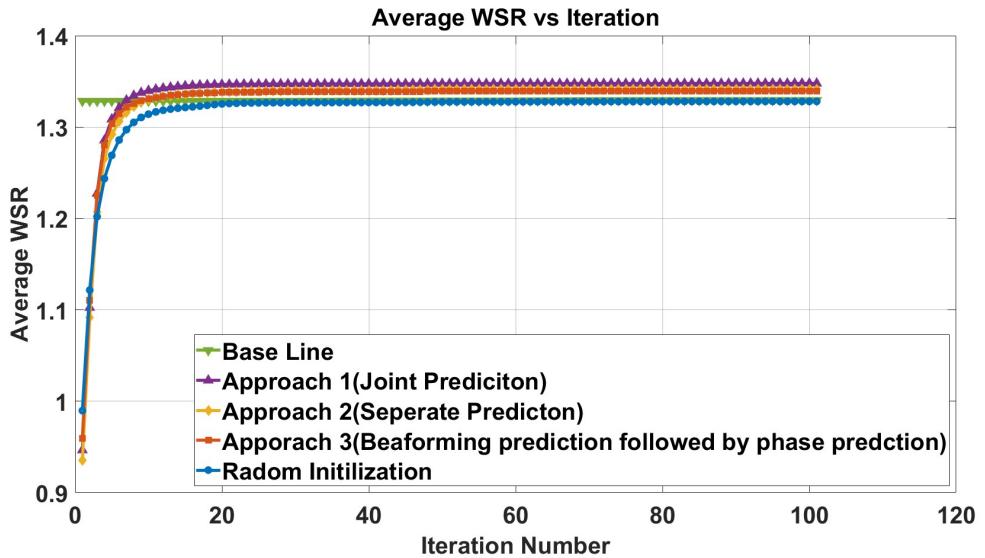


FIGURE 6.6: WSR vs. AO Iterations for $K = 1$, $N_t = 2$, $N_{\text{IRS}} = 25$ using different initialization strategies. Sequential ML initialization leads to faster convergence.

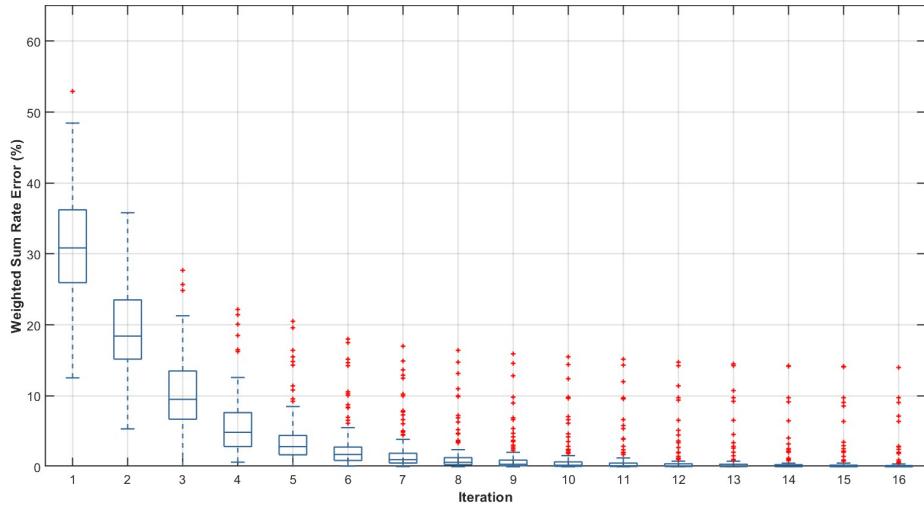


FIGURE 6.7: Box plots for Error percentage of WSR with respect to baseline optimal solution for each iteration for single mutli user scanerio(Phases and Beamforming are predicted using Approach 1).

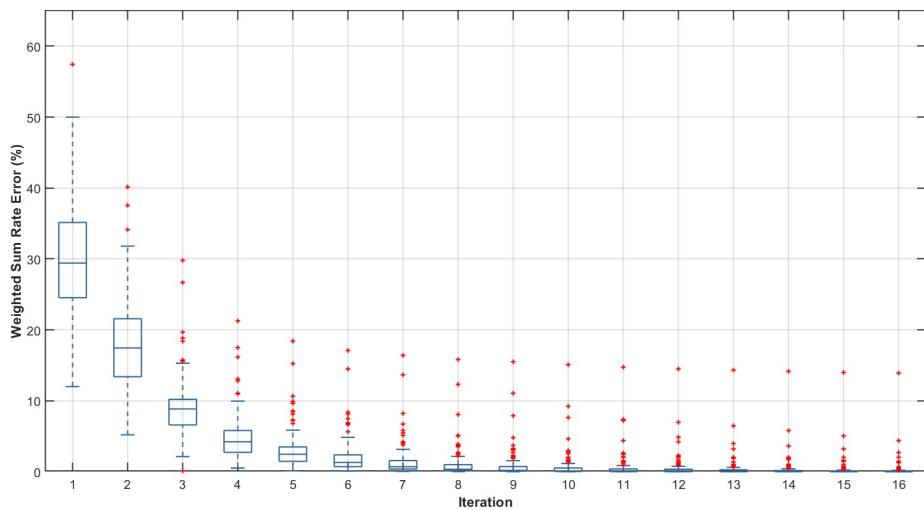


FIGURE 6.8: Box plots for Error percentage of WSR with respect to baseline optimal solution for each iteration for single mutli user scanerio (Phases and Beamforming are predicted using Approach 2).

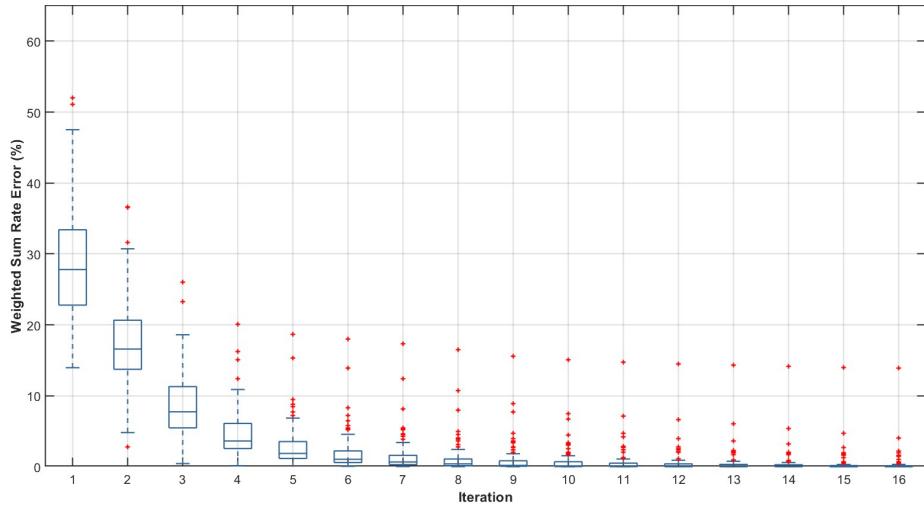


FIGURE 6.9: Box plots for Error percentage of WSR with respect to baseline optimal solution for each iteration for single mutli user scanerio (Phases and Beamforming are predicted using Approach 3).

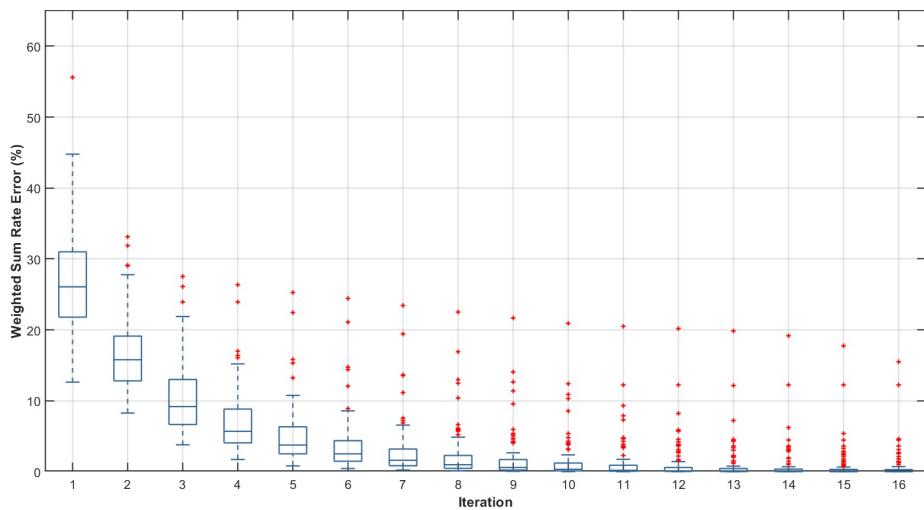


FIGURE 6.10: Box plots for Error percentage of WSR with respect to baseline optimal solution for each iteration for mutli user scanerio (Phases and Beamforming are predicted using Random Initialization).

6.3.3 Conclusion

ML-based initialization is highly effective in the single-user case, providing significant reductions in iteration count and a better starting point for optimization. However, in multi-user scenarios, the benefit of ML initialization diminishes due to the complexity of interference and the coupling of optimization variables across users. This suggests that advanced models or alternative formulations may be necessary for effective ML-based acceleration in multi-user IRS systems.

Chapter 7

Project Gradient Method: Simplification of Data and Extended Use Cases

The Projected Gradient Method (PGM) is proposed in [2] as an efficient iterative approach to solve the challenging problem of joint optimization of IRS phase vector and beamforming vector. The method is particularly motivated by the fact that projections onto the feasible sets for both the RIS phase shifts and the transmit covariance matrix can be performed very efficiently. The fundamental principle behind this algorithm is to iteratively move in the direction that maximizes the objective function by taking a step proportional to its gradient. Since this gradient step might lead to a point outside the defined feasible sets, a subsequent projection operation maps the updated variables back onto their respective feasible domains. A notable aspect of this method is that all optimization variables are updated simultaneously in each iteration, a characteristic that contrasts sharply with other common optimization techniques.

7.1 Convergence Speed and Computational Complexity of PGM

7.1.1 Convergence Speed

The Proximal Gradient Method (PGM) consistently exhibits substantially faster convergence compared to Alternating Optimization (AO). Numerical results indicate that PGM typically converges to a near-optimal achievable rate within a relatively small number of iterations. For example, in scenarios where a direct link is available, PGM rapidly achieves its optimum value. This rapid convergence is even more pronounced when the direct link is blocked, requiring only a few iterations. This swift convergence presents a critical advantage for practical systems necessitating quick adaptation.

In contrast, AO often requires a significant number of iterations to converge, particularly when the number of Reconfigurable Intelligent Surface (RIS) elements (N_{ris}) is very large. This constitutes a considerable drawback, as large RIS arrays are precisely where the technology offers the most benefit. Although AO might initially exhibit a higher achievable rate due to its initialization strategy (which involves selecting from a large set of randomly generated RIS phase shifts and optimized Q matrices), PGMs rapid convergence quickly surpasses AO in subsequent iterations.

7.1.2 Computational Complexity

The per-iteration complexity of PGM, denoted as $C_{PGM,IT}$, is primarily determined by matrix multiplications, inversions, and projection operations. This complexity can be approximated as $\mathcal{O}(2N_{ris}N_tN_r + 2N_r^2N_t + \frac{3}{2}N_t^3 + N_t^2N_r + N_tN_{ris} + 3N_{ris} + \frac{3}{2}N_t^3)$, which simplifies to $\mathcal{O}(N_{ris}N_tN_r)$ when N_{ris} is significantly larger than N_t and N_r .

7.2 System Model and Achievable Rate Optimization

7.2.1 System Architecture

We consider a wireless MIMO system with N_t transmit and N_r receive antennas, both configured as Uniform Linear Arrays (ULAs) on parallel vertical walls, separated by a distance D as shown in figure 7.1.

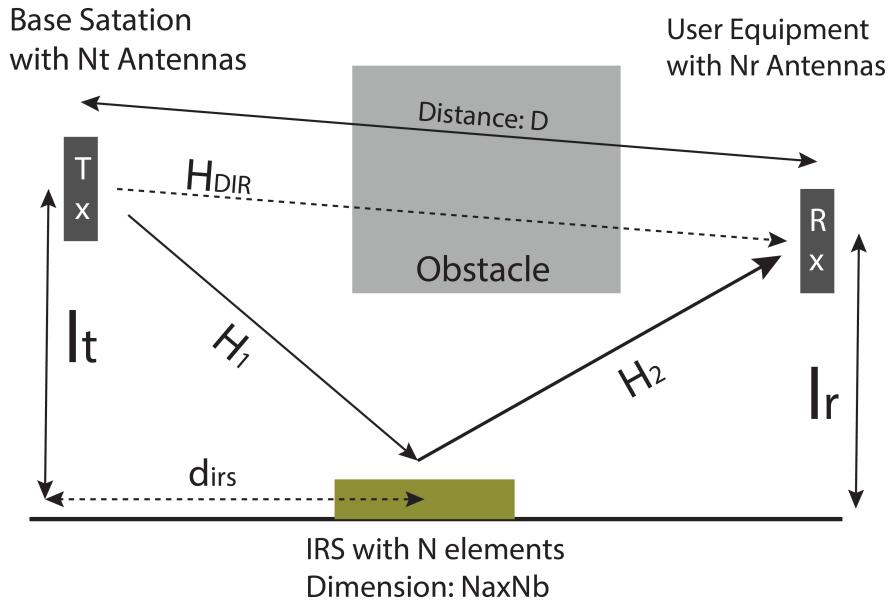


FIGURE 7.1: Aerial view of the considered system, single user, single IRS.

Due to an obstacle (e.g., a building), the direct link between the transmitter and receiver suffers attenuation. To enhance communication, a rectangular Reconfigurable Intelligent Surface (RIS) of size $a \times b$ is deployed.

The RIS consists of $N_{ris} = N_a N_b$ ideal reflecting elements, arranged in a Uniform Rectangular Array (URA). Each element has size $\lambda/2 \times \lambda/2$ and spacing $s_{ris} = \lambda/2$, and reflects with unit amplitude, i.e., $|\theta_l| = 1$.

Key distances include:

- d_{ris} : distance from RIS center to the transmit array plane,
- l_t : distance from the transmit array to the RIS plane,
- l_r : distance from the receive array to the RIS plane.

The received signal is modeled as $\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{n}$, where $\mathbf{H} \in \mathbb{C}^{N_r \times N_t}$ is the composite channel matrix, \mathbf{x} is the transmit signal, and $\mathbf{n} \sim \mathcal{CN}(\mathbf{0}, N_0\mathbf{I})$ is the noise. The transmit signal satisfies a power constraint $\text{Tr}(\mathbf{Q}) \leq P_t$, where $\mathbf{Q} = \mathbb{E}[\mathbf{x}\mathbf{x}^H]$.

The composite channel is given by:

$$\mathbf{H} = \mathbf{H}_{DIR} + \mathbf{H}_{INDIR}$$

- \mathbf{H}_{DIR} models the Rician fading direct link.
- $\mathbf{H}_{INDIR} = \beta_{INDIR}^{-1} \mathbf{H}_2 \mathbf{F}(\theta) \mathbf{H}_1$ models the RIS-assisted link, where $\mathbf{F}(\theta) = \text{diag}(\theta)$ contains the RIS phase shifts.

The channel matrices for the direct and indirect links are formulated as follows:

Direct Link Channel (H_{DIR})

The direct link channel matrix, $H_{DIR} \in \mathbb{C}^{N_r \times N_t}$, adopts the Rician fading channel model and is given by:

$$\mathbf{H}_{DIR} = \sqrt{\frac{\beta_{DIR}^{-1}}{\sqrt{K} + 1}} \left(\sqrt{K} \mathbf{H}_{D,LOS} + \mathbf{H}_{D,NLOS} \right)$$

where:

- $H_{D,LOS}(r, t) = e^{-j2\pi d_{r,t}/\lambda}$ is the Line-of-Sight (LOS) component, with $d_{r,t}$ being the distance between the t -th transmit antenna and the r -th receive antenna.

- $H_{D,NLOS}$ contains elements that are independent and identically distributed (i.i.d.) according to $\mathcal{CN}(0, 1)$.
- $\beta_{DIR} = (4\pi/\lambda)^2 d_0^{\alpha_{DIR}}$ is the Free Space Path Loss (FSPL) for the direct link, where $d_0 = \sqrt{D^2 + (l_t - l_v)^2}$ is the distance between the transmit array midpoint and the receive array midpoint.
- α_{DIR} is the path loss exponent of the direct link, influenced by the presence of an obstacle.
- K is the Rician factor.

7.2.1.1 Indirect Link Channels (H_1 and H_2)

The indirect link channel, H_{INDIR} , occurs via the Reconfigurable Intelligent Surface (RIS) and assumes a far-field model for signal transmission. It is expressed as $H_{INDIR} = \sqrt{\beta_{INDIR}^{-1}} H_2 F(\theta) H_1$.

The overall Free Space Path Loss (FSPL) for the indirect link, β_{INDIR}^{-1} , is calculated as: which simplifies to:

$$\beta_{INDIR}^{-1} = \frac{\lambda^4}{256\pi^2} \frac{(l_t/d_1 + l_r/d_2)^2}{d_1^2 d_2^2}.$$

where:

- $d_1 = \sqrt{d_{ris}^2 + l_t^2}$ is the distance between the transmit array midpoint and the RIS center.
- $d_2 = \sqrt{(D - d_{ris})^2 + l_r^2}$ is the distance between the RIS center and the receive array midpoint.
- $\cos \gamma_1 = l_t/d_1$ and $\cos \gamma_2 = l_r/d_2$ are the cosines of the angles of incidence and reflection, respectively.

according to [6], [7] the FSPL for indirect Line of sight paths is directly proportional to square of sum of angle of arrival and departure and inversely proportional to square distance between the transmitting end and receiving end.

7.2.1.2 Line of Sight Channel (H)

The channel between two nodes that are in line of sight, $H_1 \in \mathbb{C}^{N_{\text{receiving_node}} \times N_{\text{transmitting_node}}}$, also utilizes the Rician fading channel model and is given by:

$$\mathbf{H}_2 = \sqrt{\frac{1}{K+1}} \left(\sqrt{K} \mathbf{H}_{2,\text{LOS}} + \mathbf{H}_{2,\text{NLOS}} \right)$$

where:

- $H_{\text{LOS}}(l, t) = e^{-j2\pi d_{l,t}/\lambda}$ is the LOS component, with $d_{l,t}$ being the distance between the t -th transmit node and the l -th receiveing node.
- H_{NLOS} contains elements that are i.i.d. according to $\mathcal{CN}(0, 1)$.

We assume perfect Channel State Information (CSI) at both transmitter and RIS for theoretical analysis. The RIS is also assumed to operate in the far-field region.

7.2.2 Achievable Rate Optimization

The objective is to maximize the achievable rate under a transmit power constraint. Assuming Gaussian signaling and perfect CSI, the rate is given by:

$$R = \log_2 \det \left(\mathbf{I} + \frac{1}{N_0} \mathbf{H} \mathbf{Q} \mathbf{H}^H \right)$$

where \mathbf{H} depends on the RIS phases θ .

The joint optimization problem is:

$$\underset{\theta, \mathbf{Q}}{\text{maximize}} \quad \ln \det (\mathbf{I} + \mathbf{Z}(\theta) \mathbf{Q} \mathbf{Z}^H(\theta))$$

$$\text{subject to: } \text{Tr}(\mathbf{Q}) \leq P_t$$

$$\mathbf{Q} \succeq 0$$

$$|\theta_l| = 1, \quad l = 1, 2, \dots, N_{ris}$$

where $\mathbf{Z}(\theta) = \mathbf{H}_{DIR} + \mathbf{H}_2 \mathbf{F}(\theta) \mathbf{H}_1$, normalized by $1/N_0$.

This is a nonconvex optimization problem, as the objective is neither convex nor concave in θ and \mathbf{Q} . Algorithms like the Projected Gradient Method (PGM) can only guarantee convergence to a stationary point, not the global optimum. Multiple initializations or heuristic methods are often used in practice to approach high-quality solutions.

7.3 Proposed Projected Gradient Method for 1 User, 1 IRS

7.3.1 System Model

System models same as depicted in figure 7.1

7.3.2 Algorithm

Algorithm 1 PGM for 1 user, 1 IRS

Require: Initial values $\theta_0 = [1 \ 1 \ \dots \ 1]^T$, $Q_0 = (P_t/N_t)I$, and step size $\mu > 0$.

for $n = 1, 2, \dots$ **do**

Compute Gradients:

$$K(\theta_n, Q_n) = (I + Z(\theta_n)Q_nZ^H(\theta_n))^{-1}$$

$$\nabla_{\theta} f(\theta_n, Q_n) = \text{vec}_d \left(H_2^H K(\theta_n, Q_n) Z(\theta_n) Q_n \bar{H}_1^H \right) \quad \triangleright \text{W.r.t. } \theta^*$$

$$\nabla_Q f(\theta_n, Q_n) = Z^H(\theta_n) K(\theta_n, Q_n) Z(\theta_n) \quad \triangleright \text{W.r.t. } Q^+$$

Update and Project θ :

$$\theta_{n+1} = P_{\Theta}(\theta_n + \mu \nabla_{\theta} f(\theta_n, Q_n)) \quad \triangleright \text{Projection onto unit circle}$$

Where $P_{\Theta}(u)$ for $u \in \mathbb{C}^{N_{ris} \times 1}$ is defined element-wise as:

$$\bar{u}_l = \begin{cases} \frac{u_l}{|u_l|} & \text{if } u_l \neq 0 \\ e^{j\phi}, \ \phi \in [0, 2\pi] & \text{if } u_l = 0 \end{cases}, \quad l = 1, \dots, N_{ris}$$

Update and Project Q :

$$Q_{n+1} = P_Q(Q_n + \mu \nabla_Q f(\theta_n, Q_n)) \quad \triangleright \text{Projection via water-filling}$$

Where $P_Q(Y)$ uses eigen-decomposition $Y = U\Sigma U^H$ and sets $Q = UDU^H$ with:

$$d_i = (\sigma_i - \gamma)_+, \quad i = 1, \dots, N_t$$

where $\gamma \geq 0$ is chosen to meet the power constraint $\text{Tr}(Q) \leq P_t$.

end for

TABLE 7.1: Simulation Parameters for Single-User, Single IRS scenario

| Parameter | Value / Description |
|--|--|
| General System Parameters | |
| Frequency (f) | 2 GHz ($\lambda = 15$ cm) |
| Transmit Antenna Separation (s_t) | $\lambda/2 = 7.5$ cm |
| Receive Antenna Separation (s_r) | $\lambda/2 = 7.5$ cm |
| RIS Element Separation (s_{ris}) | $\lambda/2 = 7.5$ cm |
| Distance D | 500 m |
| IRS Distance (d_{irs}) | 40 m |
| Transmitter Height (l_t) | 20 m |
| Receiver Height (l_r) | 100 m |
| Number of Transmit Antennas (N_t) | 8 |
| Number of Receive Antennas (N_r) | 4 |
| Direct Path Loss Exponent (α_{DIR}) | 3 |
| Number of RIS Elements (N_{ris}) | 225 (15 × 15 grid, area > 1 m ²) |
| Rician Factor (K) | 1 |
| Total Average Transmit Power (P_t) | 0 dB |
| Noise Power (N_0) | -120 dB |
| Channel Realizations | Averaged over 200 independent realizations |

7.3.3 Results

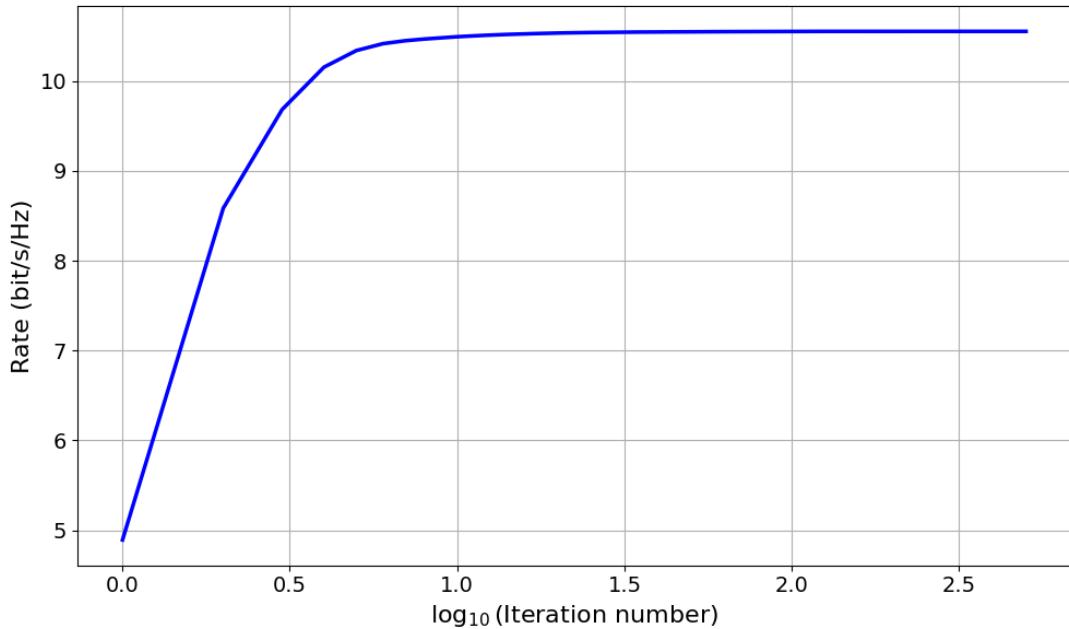


FIGURE 7.2: Achievable rate by PGM for single user single IRS

7.3.4 Effects of Quantization of Phases

The quantization of IRS (Intelligent Reflecting Surface) phases is a practical and essential step toward real-time implementation, aimed at reducing both computational complexity and communication overhead. In realistic systems, achieving and communicating continuous phase shifts with infinite precision is infeasible due to hardware constraints and the limited resolution of digital control circuits. By discretizing the phase values allowing each IRS element to take values only from a finite set we simplify the control problem and make it more suitable for real-world deployment. This quantization transforms the problem from a continuous regression task to a discrete classification problem, where each element's phase is selected from a predefined codebook. Consequently, this enables the use of classification-based machine learning models, which are often more stable and efficient for training in discrete output spaces.

However, this benefit comes at a cost. Quantization increases the size of the model significantly, especially in systems with a large number of IRS elements. Each element may require a dedicated classifier to predict its quantized phase, which scales the model parameters and computational requirements. This becomes a concern for edge devices or systems with limited processing power. Moreover, quantization introduces a trade-off between system complexity and achievable performance. As the number of quantization bits decreases, the optimization space becomes simpler and faster to explore, but this also reduces the ability of the IRS to finely tune the reflected signal, potentially leading to suboptimal performance.

Despite this, results in figure 7.3 indicate a promising middle ground. Using at least 4-bit quantization, where each phase element can take one of 16 discrete values, we observed that the system's achievable rate was nearly indistinguishable from the case with continuous (unquantized) phase values. This suggests that a relatively low-bit quantization is sufficient to retain most of the performance benefits of IRS-assisted communication, while dramatically reducing system complexity and enabling more

feasible hardware implementation. Thus, 4-bit quantization serves as a practical design choice, offering a good balance between rate performance and deployment simplicity. Future work may explore model compression, shared architectures, or low-rank approximations to further reduce model size without compromising accuracy.

7.3.4.1 Algorithm Updates

Only θ has been changes and the rest of the algorithm is same.

Input: Phase vector $\boldsymbol{\theta}$ from Algorithm 1

Compute angle (in radians): $\phi = \angle(\boldsymbol{\theta})$

Quantize: $\phi_q = \text{Quantize Angles With N bits}(\phi, N_{\text{bits}})$

Final quantized vector: $\boldsymbol{\theta} = \frac{1}{e}^{j\phi_q}$

7.3.4.2 Results

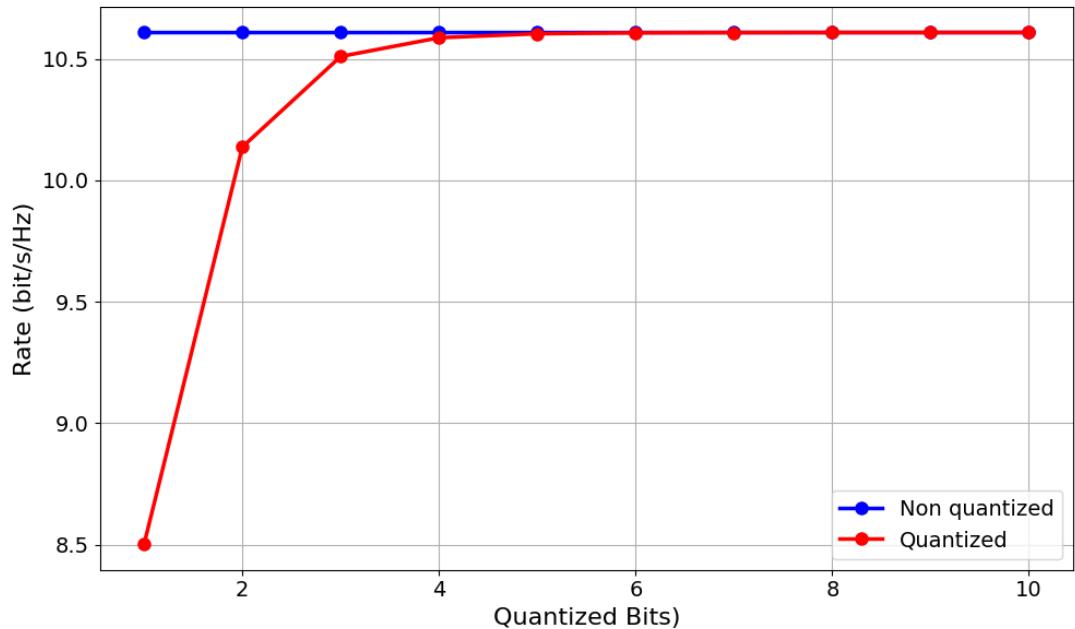


FIGURE 7.3: Achievable rate by PGM for quantized phases.

7.3.5 Effect of IRS Element Grouping

Another strategy we explored to reduce system complexity is the *grouping of IRS elements* into square patches, effectively enforcing the same phase shift across all elements within each group. This significantly reduces the overall dimensionality of the phase vector, thereby shrinking the optimization search space and making the learning and refinement processes more computationally efficient. Instead of assigning an independent phase to each of the N_{IRS} elements, we form groups of size $n_a \times n_a$, treating each group as a single control unit. As a result, the total number of independent phase values to be optimized becomes N_{IRS}/n_a^2 , drastically lowering both the number of parameters in the model and the number of target variables the machine learning model needs to predict.

From our simulation results shown in figure 7.4, it is evident that increasing the grouping size indeed leads to a substantial reduction in computational complexity. Fewer dimensions mean faster convergence, lower memory requirements, and reduced training time, all of which are desirable in practical systems with limited resources. However, this gain in efficiency comes with a cost: a significant degradation in the achievable rate, particularly when large group sizes are used. This is because the IRS loses its fine-grained control over the reflection phase profile, and the ability to adaptively steer signals with high spatial precision is compromised. Essentially, the trade-off is between model simplicity and performance fidelity a recurring theme in IRS design.

Nevertheless, this grouping approach is particularly useful when quick, approximate solutions are acceptable, such as in rapidly changing channel conditions or latency-sensitive applications. The reduction in phase vector dimensions not only lightens the ML model's prediction burden but also makes real-time hardware implementation more feasible, as fewer control signals need to be transmitted to the IRS controller. Future improvements may involve adaptive grouping, where group sizes

are dynamically selected based on channel statistics or user location, allowing the system to maintain high performance while keeping complexity in check.

7.3.5.1 Algorithm Updates

Only the gradient of θ has been changes and the rest of the algorithm is same as algorithm 1.

$$\nabla_{\theta_{\text{group}}} f(\theta_n, Q_n) = \frac{1}{n_a^2} \sum_{i \in \mathcal{G}_p} \nabla_{\theta_i} f(\theta_n, Q_n) = \frac{1}{n_a^2} \sum_{i \in \mathcal{G}_p} \text{vec}_d \left(H_2^H K(\theta_n, Q_n) Z(\theta_n) Q_n \bar{H}_1^H \right)_i$$

\mathcal{G}_p is group of patch

7.3.5.2 Results

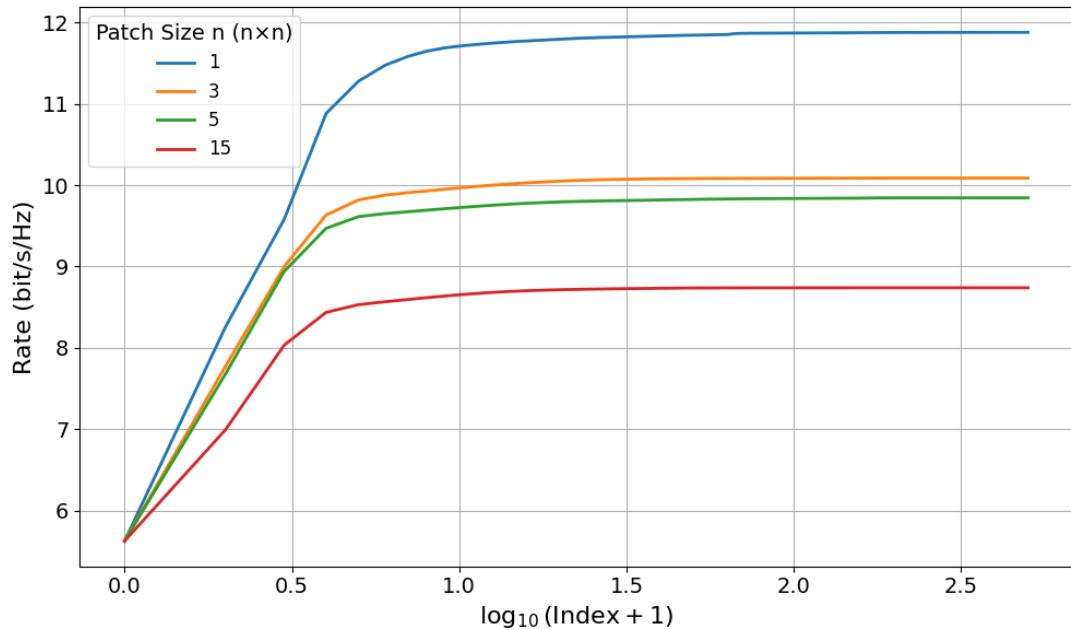


FIGURE 7.4: Achievable rate for PGM for grouping IRS elements

7.4 Proposed Projected Gradient Method for 1 User, 2 IRS

7.4.1 System Model

The channel between the two IRS is calculated using the section 7.2.1.2 and the FSPL for path transmitter, IRS 1, IRS 2, receiver is given by

$$\begin{aligned}
 d_1 &= \sqrt{(d_{\text{ris}} + d_{\text{ris,ris_x}})^2 + (d_{\text{ris,ris_y}} - l_t)^2} \\
 d_2 &= \sqrt{d_{\text{ris,ris_x}}^2 + d_{\text{ris,ris_y}}^2} \\
 d_3 &= \sqrt{(D - d_{\text{ris}})^2 + l_r^2} \\
 \beta_{\text{INDIR}}^{-1} &= \frac{\lambda^4}{256\pi^2} \frac{\left(\frac{d_{\text{ris,ris_y}} - l_t}{d_1} + \frac{d_{\text{ris,ris_y}}}{d_2}\right)^2 \left(\frac{d_{\text{ris,ris_y}}}{d_2} + \frac{l_r}{d_3}\right)^2}{(d_1 d_2 d_3)^2}
 \end{aligned}$$

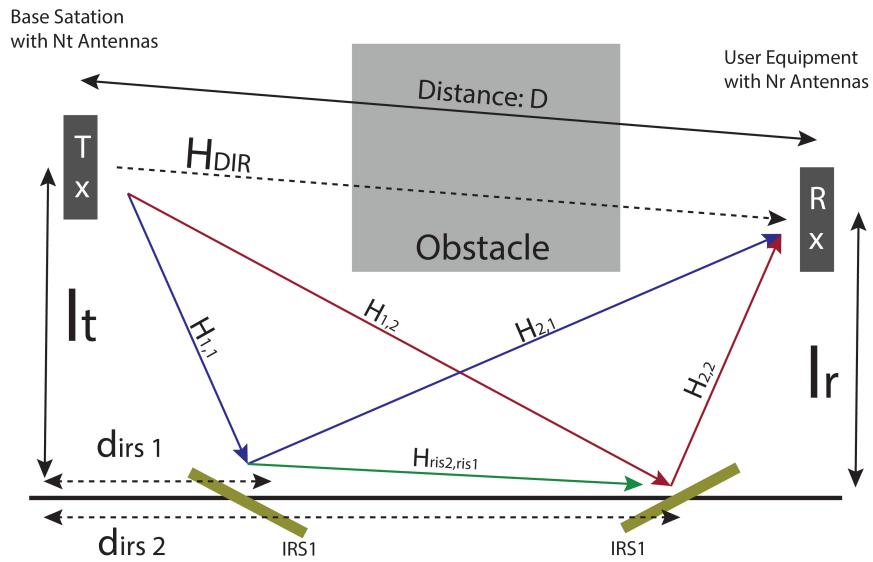


FIGURE 7.5: Aerial view of the considered system, single user, double IRS.

7.4.2 Algorithm

Algorithm 2 PGM for 1 user, 2 IRS

Require: Initial values $\theta_{0_irs1} = \theta_{0_irs2} = [1 \ 1 \ \cdots \ 1]^T$, $Q_0 = (P_t/N_t)I$, and step size $\mu > 0$.

for $n = 1, 2, \dots$ **do**

Compute Gradients:

$$\begin{aligned} Z(\theta_{n_irs1}, \theta_{n_irs2}) &= H_{DIR} + \\ &H_{2_irs1}F(\theta_{n_irs1})H_{1_irs1} + \\ &H_{2_irs2}F(\theta_{n_irs2})H_{1_irs2} + \\ &H_{2_irs2}F(\theta_{n_irs2})H_{ris2_ris1}F(\theta_{n_irs1})H_{1_irs1}, \\ &\text{normalized by } 1/N_0. \end{aligned}$$

$$K(\theta_{n_irs1}, \theta_{n_irs2}, Q_n) = (I + Z(\theta_{n_irs1}, \theta_{n_irs2})Q_nZ(\theta_{n_irs1}, \theta_{n_irs2})^H)^{-1}$$

$$\begin{aligned} \nabla_\theta f(\theta_{n_irs1}, \theta_{n_irs2}, Q_n)_{irs1} &= \\ &\text{vec}_d \left(H_{2_irs1}^H K(\theta_{n_irs1}, \theta_{n_irs2}, Q_n) Z(\theta_{n_irs1}, \theta_{n_irs2}) Q_n \bar{H}_{1_irs1}^H \right) + \\ &\text{vec}_d \left((H_{2_irs2}F(\theta_{n_irs2})H_{ris2_ris1})^H K(\theta_{n_irs1}, \theta_{n_irs2}, Q_n) Z(\theta_{n_irs1}, \theta_{n_irs2}) Q_n \bar{H}_{1_irs1}^H \right) \end{aligned}$$

$$\begin{aligned} \nabla_\theta f(\theta_n, Q_n)_{irs2} &= \\ &\text{vec}_d \left(H_{2_irs2}^H K(\theta_{n_irs2}, Q_n) Z(\theta_{n_irs1}, \theta_{n_irs2}) Q_n \bar{H}_{1_irs2}^H \right) + \\ &\text{vec}_d \left(H_{2_irs2}^H K(\theta_{n_irs2}, Q_n) Z(\theta_{n_irs1}, \theta_{n_irs2}) Q_n (H_{ris2_ris1}F(\theta_{n_irs1})\bar{H}_{1_irs1})^H \right) \end{aligned}$$

$$\nabla_Q f(\theta_{n_irs1}, \theta_{n_irs2}, Q_n) = Z^H(\theta_{n_irs1}, \theta_{n_irs2}) K(\theta_{n_irs1}, \theta_{n_irs2}, Q_n) Z(\theta_{n_irs1}, \theta_{n_irs2})$$

Update and Project θ :

$$\theta_{n+1} = P_\Theta(\theta_{n_irs1}, \theta_{n_irs2} + \mu \nabla_\theta f(\theta_{n_irs1}, \theta_{n_irs2}, Q_n)) \quad \triangleright \text{Projection onto unit circle}$$

Where $P_\Theta(u)$ for $u \in \mathbb{C}^{N_{ris} \times 1}$ is defined element-wise as:

$$\bar{u}_l = \begin{cases} \frac{u_l}{|u_l|} & \text{if } u_l \neq 0 \\ e^{j\phi}, \ \phi \in [0, 2\pi] & \text{if } u_l = 0 \end{cases}, \quad l = 1, \dots, N_{ris}$$

Update and Project Q :

$$Q_{n+1} = P_Q(Q_n + \mu \nabla_Q f(\theta_{n_irs1}, \theta_{n_irs2}, Q_n)) \quad \triangleright \text{Projection via water-filling}$$

Where $P_Q(Y)$ uses eigen-decomposition $Y = U\Sigma U^H$ and sets $Q = UDU^H$ with:

$$d_i = (\sigma_i - \gamma)_+, \quad i = 1, \dots, N_t$$

where $\gamma \geq 0$ is chosen to meet the power constraint $\text{Tr}(Q) \leq P_t$.

end for

TABLE 7.2: Simulation Parameters for Single-User, Double IRS scenario

| Parameter | Value / Description |
|--|--|
| General System Parameters | |
| Frequency (f) | 2 GHz ($\lambda = 15$ m) |
| Transmit Antenna Separation (s_t) | $\lambda/2 = 7.5$ cm |
| Receive Antenna Separation (s_r) | $\lambda/2 = 7.5$ cm |
| RIS Element Separation (s_{ris}) | $\lambda/2 = 7.5$ cm |
| Distance D | 500 m |
| IRS 1 Distance (d_{irs}) | 40 m |
| IRS 2 Distance (d_{irs}) | 460 m |
| Transmitter Height (l_t) | 20 m |
| Receiver Height (l_r) | 100 m |
| Number of Transmit Antennas (N_t) | 8 |
| Number of Receive Antennas (N_r) | 4 |
| Direct Path Loss Exponent (α_{DIR}) | 3 |
| Number of RIS Elements (N_{ris}) | 225 (15 × 15 grid, area > 1 m ²) |
| Rician Factor (K) | 1 |
| Total Average Transmit Power (P_t) | 0 dB |
| Noise Power (N_0) | -120 dB |
| Channel Realizations | Averaged over 200 independent realizations |

7.4.3 Results

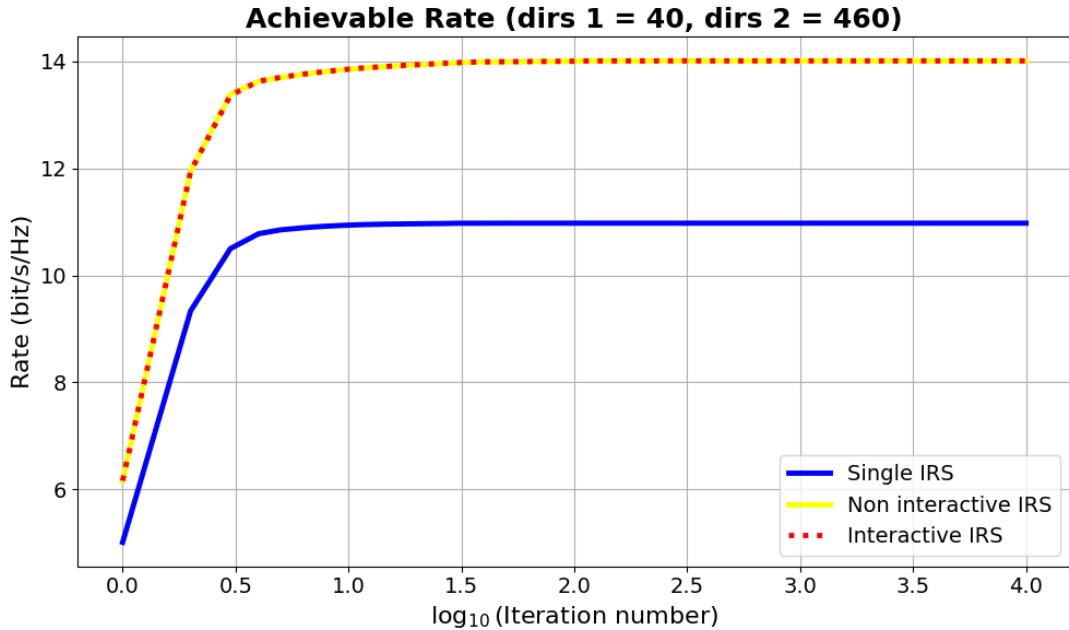


FIGURE 7.6: Achievable rate by PGM for single user, doubler IRS

7.5 Proposed Projected Gradient Method for 2 User, 1 IRS

7.5.1 System Model

We are using successive interference cancellation at user side for symbol detection. For the simplicity we are using only 1 antenna at base stations and for both the users.

$$X = X_1 + X_2$$

X_1 is system for user 1 and X_2 is symbol for user 2.

$$h_1 = (H_{\text{dir},\text{user}1} + H_{2,\text{user}1}\text{diag}(\theta_n)H_1)$$

$$h_2 = (H_{\text{dir},\text{user}2} + H_{2,\text{user}2}\text{diag}(\theta_n)H_1)$$

The rate R is calculated as follows:

If $|h_1| < |h_2|$:

$$R = \log_2 \left(1 + \frac{P_2|h_2|^2}{\sigma^2} \right) + \log_2 \left(1 + \frac{P_1|h_1|^2}{P_2|h_2|^2 + \sigma^2} \right)$$

Else (If $|h_1| \geq |h_2|$) :

$$R = \log_2 \left(1 + \frac{P_1|h_1|^2}{\sigma^2} \right) + \log_2 \left(1 + \frac{P_2|h_2|^2}{P_1|h_1|^2 + \sigma^2} \right)$$

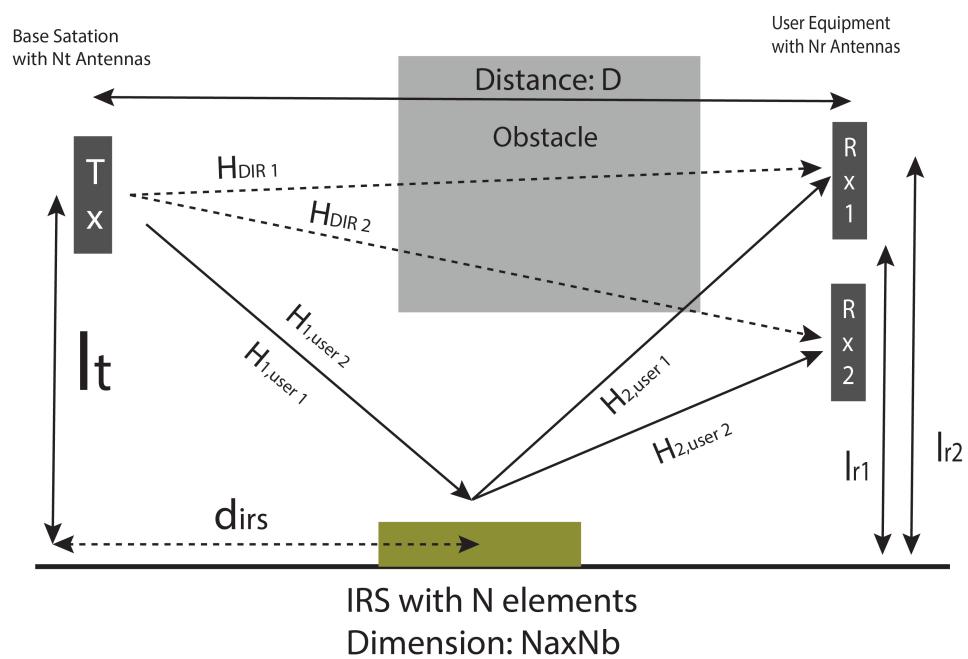


FIGURE 7.7: Aerial view of the considered system, double user, single IRS.

7.5.2 Algorithm

Algorithm 3 Projected Gradient Method (PGM) for 2 Users with 1 IRS
(Part 1)

```

1: Input: Initial values  $\theta_0 = [1 \ 1 \ \dots \ 1]^T$ ,  $P_1 = P_2 = P_t/2$ , step size  $\mu > 0$ 
2: Noation = In  $Var_j$   $Var_{i,j}$ , j = user number
3: for  $n = 1, 2, \dots$  do
4:   Compute Effective Channels:
5:    $h_1 = (\mathbf{H}_{\text{dir},1} + \mathbf{H}_{2,1} \cdot \text{diag}(e^{j\theta_n}) \cdot \mathbf{H}_{1,1})$ 
6:    $h_2 = (\mathbf{H}_{\text{dir},2} + \mathbf{H}_{2,2} \cdot \text{diag}(e^{j\theta_n}) \cdot \mathbf{H}_{1,2})$ 
7:   if  $|h_1| > |h_2|$  then
8:     Compute Gradients (Case 1):
9:      $\nabla_{P_1} = \frac{1}{\ln(2)} \left( \frac{-P_2|h_2|^4}{(|h_2|^2(P_2+P_1)+\sigma^2)(P_1|h_2|^2+\sigma^2)} + \frac{|h_1|^2}{P_1|h_1|^2+\sigma^2} \right)$ 
10:     $\nabla_{P_2} = \frac{1}{\ln(2)} \cdot \frac{|h_2|^2}{|h_2|^2(P_2+P_1)+\sigma^2}$ 
11:     $\nabla_\theta = \frac{1}{\ln(2)} \left[ \frac{P_2 \cdot \sigma^2}{(|h_1|^2(P_1+P_2)+\sigma^2)(P_1|h_1|^2+P_2|h_1|^2+\sigma^2)} \cdot 2\text{Re}((\mathbf{H}_{2,1} \odot \mathbf{H}_{1,1}) \odot \text{conj}(h_1)) \right.$ 
12:     $\left. + \frac{P_1}{(P_1|h_2|^2+\sigma^2)} \cdot 2\text{Re}((\mathbf{H}_{2,2} \odot \mathbf{H}_{1,2}) \odot \text{conj}(h_2)) \right]$ 
13:   else
14:     Compute Gradients (Case 2):
15:      $\nabla_{P_1} = \frac{1}{\ln(2)} \cdot \frac{|h_1|^2}{|h_1|^2(P_1+P_2)+\sigma^2}$ 
16:      $\nabla_{P_2} = \frac{1}{\ln(2)} \left( \frac{-P_1|h_1|^4}{(|h_1|^2(P_1+P_2)+\sigma^2)(P_2|h_2|^2+\sigma^2)} + \frac{|h_2|^2}{P_2|h_2|^2+\sigma^2} \right)$ 
17:      $\nabla_\theta = \frac{1}{\ln(2)} \left[ \frac{P_1 \cdot \sigma^2}{(|h_2|^2(P_1+P_2)+\sigma^2)(P_1|h_2|^2+P_2|h_2|^2+\sigma^2)} \cdot 2\text{Re}((\mathbf{H}_{2,2} \odot \mathbf{H}_{1,2}) \odot \text{conj}(h_2)) \right.$ 
18:      $\left. + \frac{P_2}{(P_2|h_1|^2+\sigma^2)} \cdot 2\text{Re}((\mathbf{H}_{2,1} \odot \mathbf{H}_{1,1}) \odot \text{conj}(h_1)) \right]$ 
19:   end if
20:   Update and Project  $\theta$ :
21:    $\theta_{n+1} = P_\Theta(\theta_n + \mu \nabla_\theta)$  ▷ Projection onto unit circle
22:   Where  $P_\Theta(u)$  for  $u \in \mathbb{C}^{N_{ris} \times 1}$  is defined element-wise as:

$$\bar{u}_l = \begin{cases} \frac{u_l}{|u_l|} & \text{if } u_l \neq 0 \\ e^{j\phi}, \ \phi \in [0, 2\pi] & \text{if } u_l = 0 \end{cases}, \quad l = 1, \dots, N_{ris}$$

23: end for

```

Algorithm 4 Projected Gradient Method (PGM) for 2 Users with 1 IRS
(Part 2)

```
1: for  $n = 1, 2, \dots$  do
2:   Update and Project  $P_1$  &  $P_2$ :
3:   Compute intermediate values:
4:    $P_1 \leftarrow \frac{P_1^{(n+1)} - P_2^{(n+1)} + P_t}{2}$ 
5:    $P_2 \leftarrow P_t - P_1$ 
6:   if  $P_1 > 0$  and  $P_2 > 0$  or  $P_1 < 0$  and  $P_2 < 0$  then
7:      $P_1^{(n+1)} \leftarrow |P_1|$ 
8:      $P_2^{(n+1)} \leftarrow |P_2|$ 
9:   else if  $P_1 \leq 0$  then
10:     $P_1^{(n+1)} \leftarrow 0$ 
11:     $P_2^{(n+1)} \leftarrow P_t$ 
12:   else if  $P_2 \leq 0$  then
13:     $P_1^{(n+1)} \leftarrow P_t$ 
14:     $P_2^{(n+1)} \leftarrow 0$ 
15:   end if
16: end for
```

TABLE 7.3: Simulation Parameters for Double-User, Single IRS scenario

| Parameter | Value / Description |
|--|--|
| General System Parameters | |
| Frequency (f) | 2 GHz ($\lambda = 15$ m) |
| Transmit Antenna Separation (s_t) | $\lambda/2 = 7.5$ cm |
| Receive Antenna Separation (s_r) | $\lambda/2 = 7.5$ cm |
| RIS Element Separation (s_{ris}) | $\lambda/2 = 7.5$ cm |
| Distance D for Rx_1 | 500 m |
| Distance D for Rx_2 | 500 m |
| IRS 1 Distance (d_{irs}) | 40 m |
| Transmitter Height (l_t) | 20 m |
| Receiver Height 1 (l_{Rx1}) | 100 m |
| Receiver Height 2 (l_{Rx2}) | 40 m |
| Number of Transmit Antennas (N_t) | 1 |
| Number of Receive Antennas (N_r) | 1 |
| Direct Path Loss Exponent (α_{DIR}) | 3 |
| Number of RIS Elements (N_{ris}) | 225 (15 × 15 grid, area > 1 m ²) |
| Rician Factor (K) | 1 |
| Total Average Transmit Power (P_t) | 0 dB |
| Noise Power (N_0) | -120 dB |
| Channel Realizations | Averaged over 200 independent realizations |

7.5.3 Results

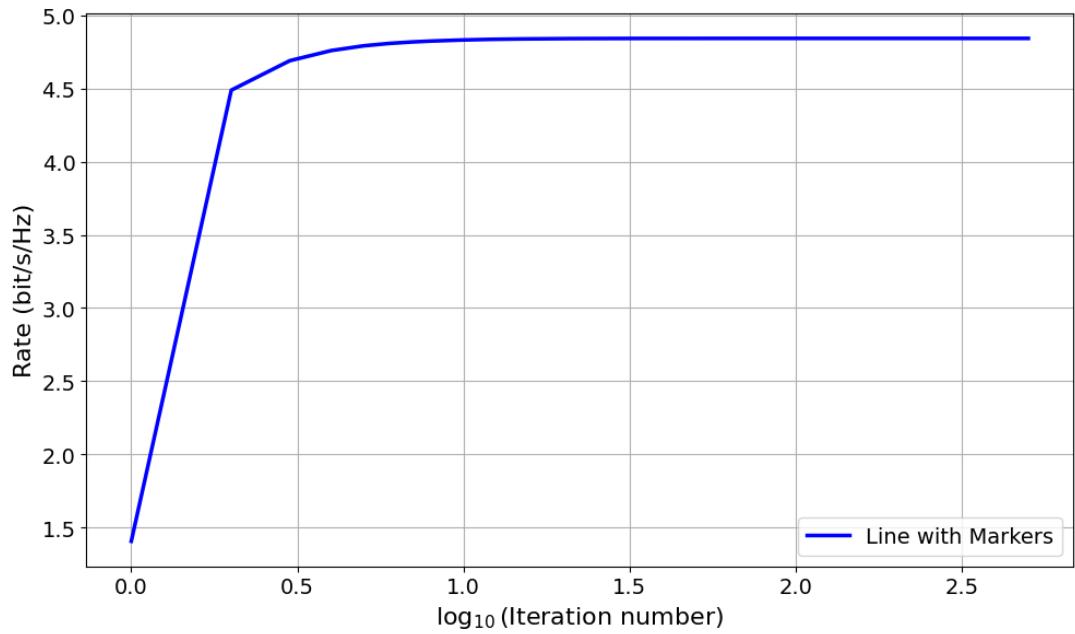


FIGURE 7.8: Achievable rate by PGM for double user, single IRS

7.6 Proposed Projected Gradient Method for 2 User, 2 IRS

7.6.1 System Model

We are using successive interference cancellation at user side for symbol detection. For the simplicity we are using only 1 antenna at base stations and for both the users.

$$X = X_1 + X_2$$

X_1 is system for user 1 and X_2 is symbol for user 2.

$$\begin{aligned} h_1(\theta_{n.irs1}, \theta_{n.irs2}) &= H_{DIR,user1} + H_{2.irs1,user1}F(\theta_{n.irs1})H_{1.irs1,user1} + \\ &\quad H_{2.irs2,user1}F(\theta_{n.irs2})H_{1.irs2,user1} + \\ &\quad H_{2.irs2,user1}F(\theta_{n.irs2})H_{ris2.ris1}F(\theta_{n.irs1})H_{1.irs1,user1}, \\ h_2(\theta_{n.irs1}, \theta_{n.irs2}) &= H_{DIR,user2} + H_{2.irs1,user2}F(\theta_{n.irs1})H_{1.irs1,user2} + \\ &\quad H_{2.irs2,user2}F(\theta_{n.irs2})H_{1.irs2,user2} + \\ &\quad H_{2.irs2,user2}F(\theta_{n.irs2})H_{ris2.ris1}F(\theta_{n.irs1})H_{1.irs1,user2}, \end{aligned}$$

The rate R is calculated as follows:

If $|h_1| < |h_2|$:

$$R = \log_2 \left(1 + \frac{P_2|h_2|^2}{\sigma^2} \right) + \log_2 \left(1 + \frac{P_1|h_1|^2}{P_2|h_2|^2 + \sigma^2} \right)$$

Else (If $|h_1| \geq |h_2|$) :

$$R = \log_2 \left(1 + \frac{P_1|h_1|^2}{\sigma^2} \right) + \log_2 \left(1 + \frac{P_2|h_2|^2}{P_1|h_2|^2 + \sigma^2} \right)$$

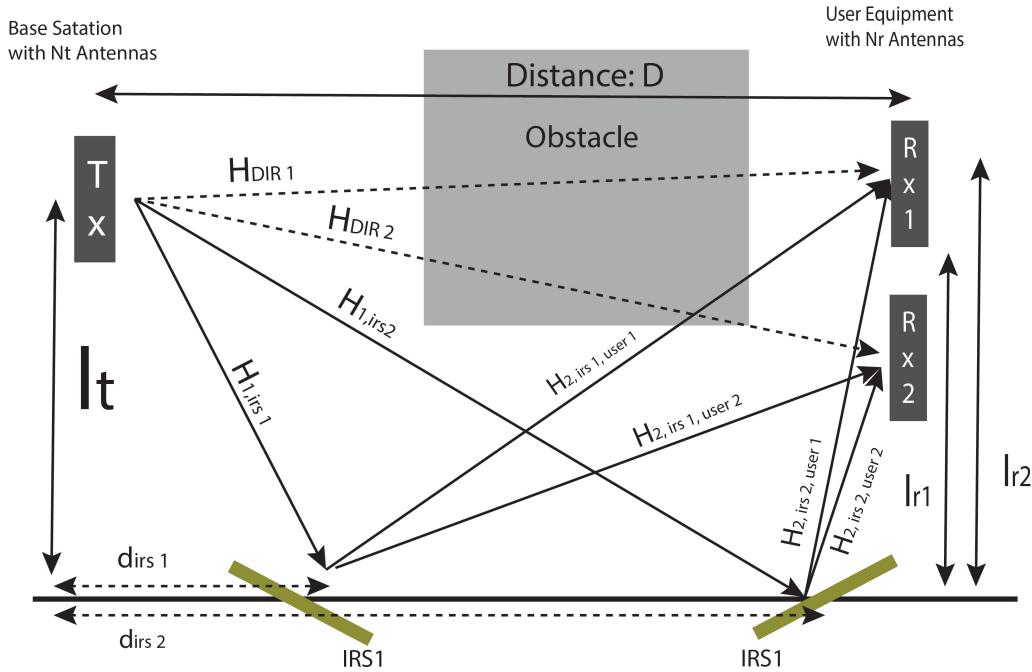


FIGURE 7.9: Aerial view of the considered system, double user, double IRS.

7.6.2 Algorithm

Algorithm 5 Projected Gradient Method (PGM) for 2 Users with 1 IRS
(Simplified)

- 1: **Input:** Initial values $\theta_{0,irs1} = \theta_{0,irs2} = [1 \ 1 \ \cdots \ 1]^T$, $P_1 = P_2 = P_t/2$, step size $\mu > 0$
- 2: **Notation:** $Var_{i,j,k}$: i =variable type, j =IRS number, k =user number
- 3: **for** $n = 1, 2, \dots$ **do**
- 4: Compute effective channels: h_1, h_2
- 5: Define: $\gamma_1 = |h_1|^2, \gamma_2 = |h_2|^2$
- 6: Define common terms:

$$D_1 = \ln(2), \quad \sigma^2 = \sigma^2$$

$$C_1 = \gamma_2(P_1 + P_2) + \sigma^2, \quad C_2 = \gamma_1(P_1 + P_2) + \sigma^2$$

$$D_{P1} = P_1\gamma_2 + \sigma^2, \quad D_{P2} = P_2\gamma_1 + \sigma^2$$

- 7: **if** $\gamma_1 > \gamma_2$ **then**
- 8: **Case 1:**
- 9: $\nabla_{P_1} = \frac{1}{D_1} \left(\frac{-P_2\gamma_2^2}{C_1 \cdot D_{P1}} + \frac{\gamma_1}{P_1\gamma_1 + \sigma^2} \right)$
- 10: $\nabla_{P_2} = \frac{1}{D_1} \cdot \frac{\gamma_2}{C_1}$
- 11: Compute ∇_{θ_1} and ∇_{θ_2} using:

Let $G_{1,u} = \mathbf{H}_{2,1,u} \odot \mathbf{H}_{1,1,u} + \mathbf{H}_{2,2,u} \cdot \text{diag}(\theta_{n.irs2}) \mathbf{H}_{ris2,ris1} \odot \mathbf{H}_{1,1,u}$

Let $G_{2,u} = \mathbf{H}_{2,2,u} \odot \mathbf{H}_{1,2,u} + \mathbf{H}_{2,2,u} \odot \mathbf{H}_{ris2,ris1} \cdot \text{diag}(\theta_{n.irs1}) \mathbf{H}_{1,1,u}$

- 12: $\nabla_{\theta_1} = \frac{2}{D_1} \left(\frac{P_2\sigma^2}{C_2(P_1\gamma_1 + P_2\gamma_1 + \sigma^2)} \Re(G_{1,1} \odot \text{conj}(h_1)) + \frac{P_1}{P_1\gamma_2 + \sigma^2} \Re(G_{1,2} \odot \text{conj}(h_2)) \right)$
 - 13: $\nabla_{\theta_2} = \frac{2}{D_1} \left(\frac{P_2\sigma^2}{C_2(P_1\gamma_1 + P_2\gamma_1 + \sigma^2)} \Re(G_{2,1} \odot \text{conj}(h_1)) + \frac{P_1}{P_1\gamma_2 + \sigma^2} \Re(G_{2,2} \odot \text{conj}(h_2)) \right)$
 - 14: **else**
 - 15: **Case 2:**
 - 16: $\nabla_{P_1} = \frac{1}{D_1} \cdot \frac{\gamma_1}{C_2}$
 - 17: $\nabla_{P_2} = \frac{1}{D_1} \left(\frac{-P_1\gamma_1^2}{C_2(P_2\gamma_1 + \sigma^2)} + \frac{\gamma_2}{P_2\gamma_2 + \sigma^2} \right)$
 - 18: Use the same $G_{1,u}$ and $G_{2,u}$ expressions
 - 19: $\nabla_{\theta_1} = \frac{2}{D_1} \left(\frac{P_1\sigma^2}{C_1(P_1\gamma_2 + P_2\gamma_2 + \sigma^2)} \Re(G_{1,2} \odot \text{conj}(h_1)) + \frac{P_2}{P_2\gamma_1 + \sigma^2} \Re(G_{1,1} \odot \text{conj}(h_2)) \right)$
 - 20: $\nabla_{\theta_2} = \frac{2}{D_1} \left(\frac{P_1\sigma^2}{C_1(P_1\gamma_2 + P_2\gamma_2 + \sigma^2)} \Re(G_{2,2} \odot \text{conj}(h_1)) + \frac{P_2}{P_2\gamma_1 + \sigma^2} \Re(G_{2,1} \odot \text{conj}(h_2)) \right)$
 - 21: **end if**
 - 22: **end for**
-

Algorithm 6 Projected Gradient Method (PGM) for 2 Users with 1 IRS
 (Part 2)

```

1: for  $n = 1, 2, \dots$  do
2:   Update and Project  $\theta$ :
3:      $\theta_{n+1} = P_\Theta(\theta_n + \mu \nabla_\theta)$  ▷ Projection onto unit circle
4:     Where  $P_\Theta(u)$  for  $u \in \mathbb{C}^{N_{ris} \times 1}$  is defined element-wise as:

$$\bar{u}_l = \begin{cases} \frac{u_l}{|u_l|} & \text{if } u_l \neq 0 \\ e^{j\phi}, \phi \in [0, 2\pi] & \text{if } u_l = 0 \end{cases}, \quad l = 1, \dots, N_{ris}$$

5:   Update and Project  $P_1$  &  $P_2$ :
6:   Compute intermediate values:
7:      $P_1 \leftarrow \frac{P_1^{(n+1)} - P_2^{(n+1)} + P_t}{2}$ 
8:      $P_2 \leftarrow P_t - P_1$ 
9:   if  $P_1 > 0$  and  $P_2 > 0$  or  $P_1 < 0$  and  $P_2 < 0$  then
10:     $P_1^{(n+1)} \leftarrow |P_1|$ 
11:     $P_2^{(n+1)} \leftarrow |P_2|$ 
12:   else if  $P_1 \leq 0$  then
13:     $P_1^{(n+1)} \leftarrow 0$ 
14:     $P_2^{(n+1)} \leftarrow P_t$ 
15:   else if  $P_2 \leq 0$  then
16:     $P_1^{(n+1)} \leftarrow P_t$ 
17:     $P_2^{(n+1)} \leftarrow 0$ 
18:   end if
19: end for

```

TABLE 7.4: Simulation Parameters for Double-User, Double IRS scenario

| Parameter | Value / Description |
|--|--|
| General System Parameters | |
| Frequency (f) | 2 GHz ($\lambda = 15$ m) |
| Transmit Antenna Separation (s_t) | $\lambda/2 = 7.5$ cm |
| Receive Antenna Separation (s_r) | $\lambda/2 = 7.5$ cm |
| RIS Element Separation (s_{ris}) | $\lambda/2 = 7.5$ cm |
| Distance D for Rx_1 | 500 m |
| Distance D for Rx_2 | 500 m |
| IRS 1 Distance (d_{irs}) | 40 m |
| IRS 2 Distance (d_{irs}) | 460 m |
| Transmitter Height (l_t) | 20 m |
| Receiver Height 1 (l_{Rx1}) | 100 m |
| Receiver Height 2 (l_{Rx2}) | 40 m |
| Number of Transmit Antennas (N_t) | 1 |
| Number of Receive Antennas (N_r) | 1 |
| Direct Path Loss Exponent (α_{DIR}) | 3 |
| Number of RIS Elements (N_{ris}) | 225 (15 × 15 grid, area > 1 m ²) |
| Rician Factor (K) | 1 |
| Total Average Transmit Power (P_t) | 0 dB |
| Noise Power (N_0) | -120 dB |
| Channel Realizations | Averaged over 200 independent realizations |

7.6.3 Results

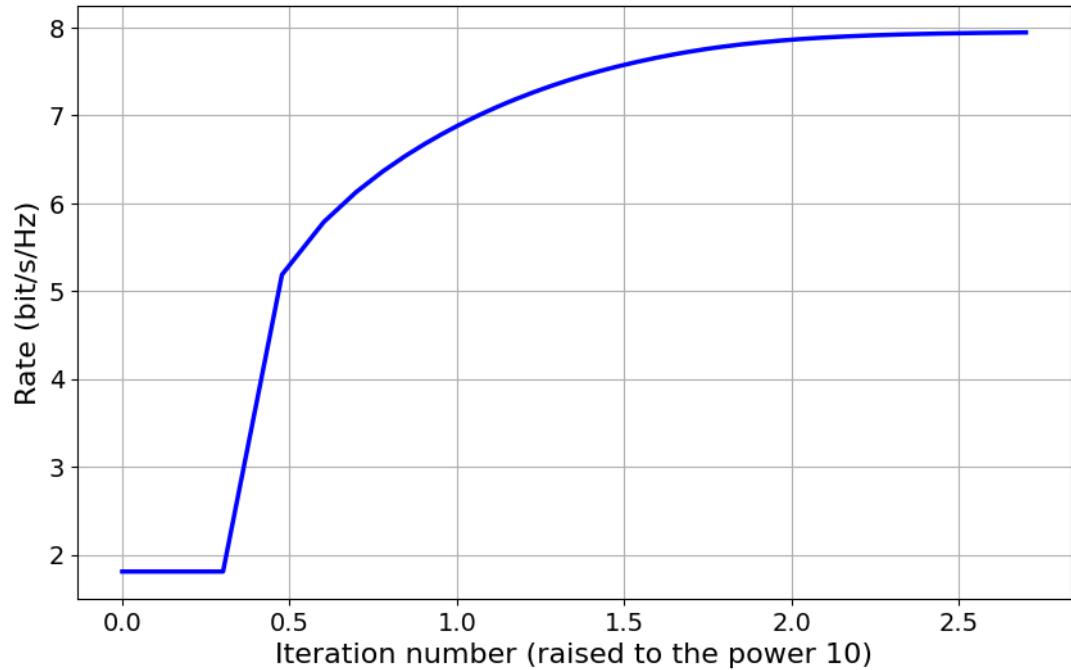


FIGURE 7.10: Achievable rate by PGM for double user, doubler IRS.

Chapter 8

Observations and Future work

8.1 Observations

1. **Impact of IRS Elements on Achievable Rate:** The simulation results demonstrate that increasing the number of Intelligent Reflecting Surface (IRS) elements leads to a noticeable enhancement in the achievable data rate. However, this improvement comes at the cost of increased computational complexity, especially in the optimization phase.
2. **Performance in Multi-IRS Setup:** When employing a multi-IRS architecture specifically where a single dominant path connects the Base Station (BS), IRS1, IRS2, and the user the achievable rate significantly increases compared to the single-IRS case. However, when additional paths are introduced (e.g., BS–IRS1–User and BS–IRS2–User), the overall rate improves, but the gain is lower than the ideal single-path scenario (BS–IRS1–IRS2–User).
3. **Redundancy in IRS1–IRS2 Path:** In simulations that considered multi-path models, both scenarios where the IRS1–IRS2 path is active and where it is ignored yielded identical achievable rates. This implies that the IRS1–IRS2

link contributes little to no benefit in such topologies. Therefore, for computational efficiency, the IRS1–IRS2 path can be omitted without performance loss.

4. **Multi-User Scenario with SIC:** In the multi-user case, using Successive Interference Cancellation (SIC) consistently led to an increase in the sum-rate, validating the theoretical benefits of SIC in IRS-assisted systems.
5. **Quantization and Dimensionality Reduction:** Applying phase quantization and grouping of IRS elements reduces the search space dimensionality, making the problem more tractable for machine learning models. Nevertheless, this reduction introduces a trade-off, as it leads to a slight drop in the achievable rate due to the constrained solution space.

8.2 Future work

- **Trade-off Between Accuracy and Complexity:** The study highlights a fundamental trade-off: while phase quantization and element grouping significantly reduce computational burden and make machine learning prediction feasible, they also limit the maximum achievable rate. This necessitates hybrid approaches that balance performance and efficiency.
- **Role of Machine Learning in Phase Prediction:** The results indicate that machine learning models can effectively predict IRS phase shifts when the problem is converted from regression to classification using quantized phase values. This simplification enables the use of classification algorithms, which are generally more robust and easier to train than regression models in high-dimensional spaces.
- **Hybrid Optimization Strategies:** A promising direction is to combine machine learning and traditional optimization. For example, ML can be used to

provide an initial estimate of the phase vector, which is then refined using a few iterations of a gradient-based optimization algorithm. This approach leverages the speed of ML and the precision of optimization, providing a balance between complexity and performance.

- **Opportunities for Adaptive Algorithms:** Future work can explore adaptive dimensionality reduction strategies where the quantization level and grouping of IRS elements are dynamically adjusted based on system requirements such as latency constraints or environment variability. Such algorithms would offer better scalability and real-time performance.
- **IRS Placement Optimization:** Another potential direction is to develop learning-based models or heuristics to optimize the physical placement of IRS elements. By strategically positioning IRSs to limit non-beneficial reflection paths and maximize coverage, one can improve the overall system efficiency and achievable rate.

References

- [1] S. Gong, X. Lu, D. T. Hoang, D. Niyato, L. Shu, D. I. Kim, and Y.-C. Liang, “Toward smart wireless communications via intelligent reflecting surfaces: A contemporary survey,” *IEEE Communications Surveys Tutorials*, vol. 22, no. 4, pp. 2283–2314, 2020.
- [2] N. S. Perović, L.-N. Tran, M. Di Renzo, and M. F. Flanagan, “Achievable rate optimization for mimo systems with reconfigurable intelligent surfaces,” *IEEE Transactions on Wireless Communications*, vol. 20, no. 6, pp. 3865–3882, 2021.
- [3] A. M. Elbir and K. V. Mishra, “A survey of deep learning architectures for intelligent reflecting surfaces,” 2022.
- [4] H. Guo, Y.-C. Liang, J. Chen, and E. G. Larsson, “Weighted sum-rate maximization for reconfigurable intelligent surface aided wireless networks,” *IEEE Transactions on Wireless Communications*, vol. 19, no. 5, pp. 3064–3076, 2020.
- [5] B. Zheng, C. You, and R. Zhang, “Double-irs assisted multi-user mimo: Cooperative passive beamforming design,” *IEEE Transactions on Wireless Communications*, vol. 20, no. 7, pp. 4513–4526, 2021.
- [6] S. W. Ellingson, “Path loss in reconfigurable intelligent surface-enabled channels,” in *2021 IEEE 32nd Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, IEEE, Sept. 2021.
- [7] S. J. Orfanidis, *Electromagnetic Waves Antennas*. New Brunswick, NJ, USA: Rutgers Univ., 2002.

- [8] S. Gong, J. Lin, J. Zhang, D. Niyato, D. I. Kim, and M. Guizani, “Optimization-driven machine learning for intelligent reflecting surfaces assisted wireless networks,” 2020.