A GNURadio Framework for Real-Time Beam Steering with Reconfigurable Intelligent Surfaces

Parker Wilmoth, George Sklivanitis, and Dimitris A. Pados

Center for Connected Autonomy and Artificial Intelligence (CA-AI.fau.edu)
Dept. of Electrical Engineering and Computer Science
Florida Atlantic University
Boca Raton, FL 33431, USA

GRCon25
Everett, Washington

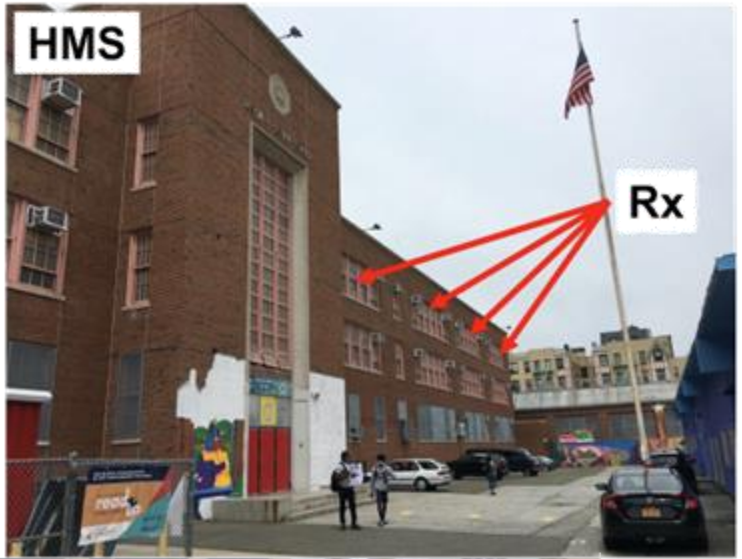
FLORIDA ATLANTIC UNIVERSITY
Center for Connected Autonomy and Artificial Intelligence
College of Engineering & Computer Science

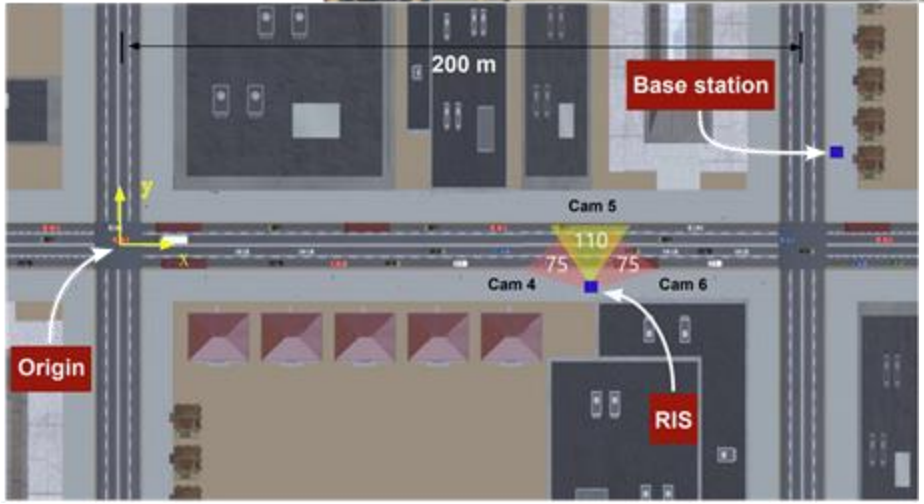
NSF
CENTER FOR SMART STREETSCAPES

mmWave Motivation & Challenges

- Deployment of 28 GHz mmWave networks in urban areas with low internet access could help improve connectivity and bridge the digital divide
- Models show that data rates > 2.8 Gb/s are achievable for at least 90% of indoor users in typical public-school buildings with lightpole BS deployments at distances up to 68 m away [1]

- **High path loss**: Signals attenuate quickly over distance, requiring more power and precise beamforming
- **Highly directional nature**: Narrow beams lead to potential misalignment, making beamforming complex
- **Dynamic channel conditions**: Mobility, obstructions, and interference cause continuous changes in signal quality

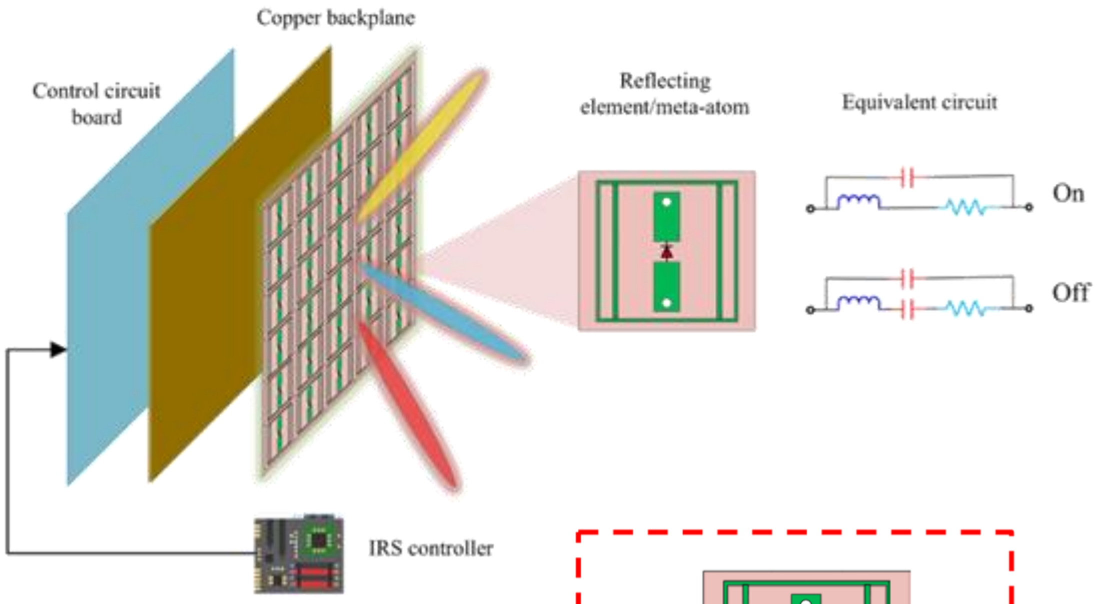


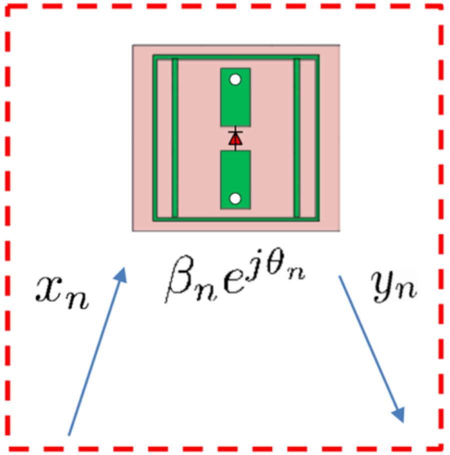


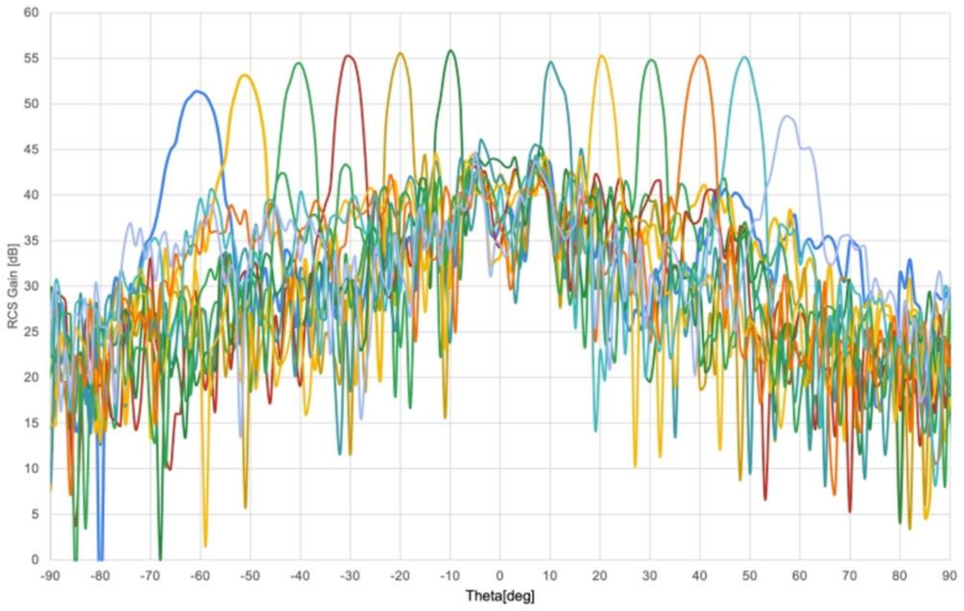
[1] M. Kohli, A. Adhikari, G. Avci, et al., Outdoor-to-indoor 28 GHz Wireless Measurements in Manhattan: Path Loss, Location Impacts, and 90% Coverage, *in Proceedings of the 23rd MobiHoc*, Association for Computing Machinery, pp. 201–210. 2022.

What is a RIS?

- Digitally controllable scatters
- A digitally-controlled metasurface with massive low-cost passive reflecting
- **Random and uncontrollable** propagation environment into a smart radio environment (**controllable and programmable**)
- The RIS is neither part of the transmitter nor the receiver
- Reflect signals to a desired direction.





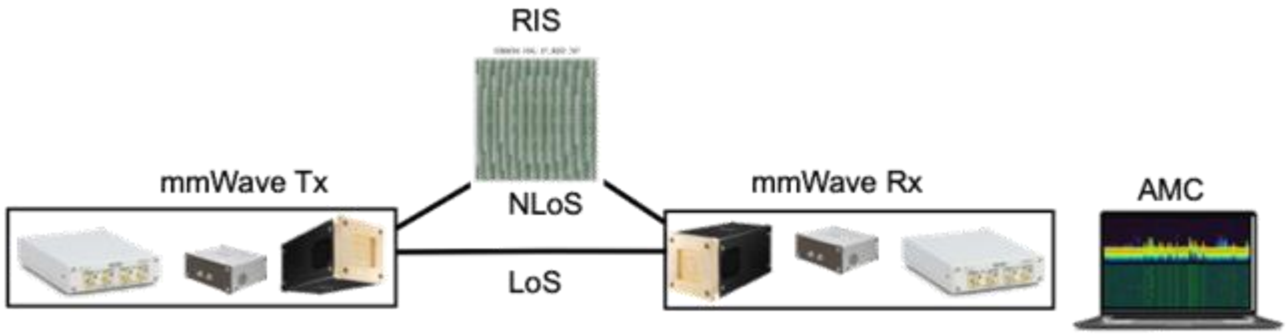


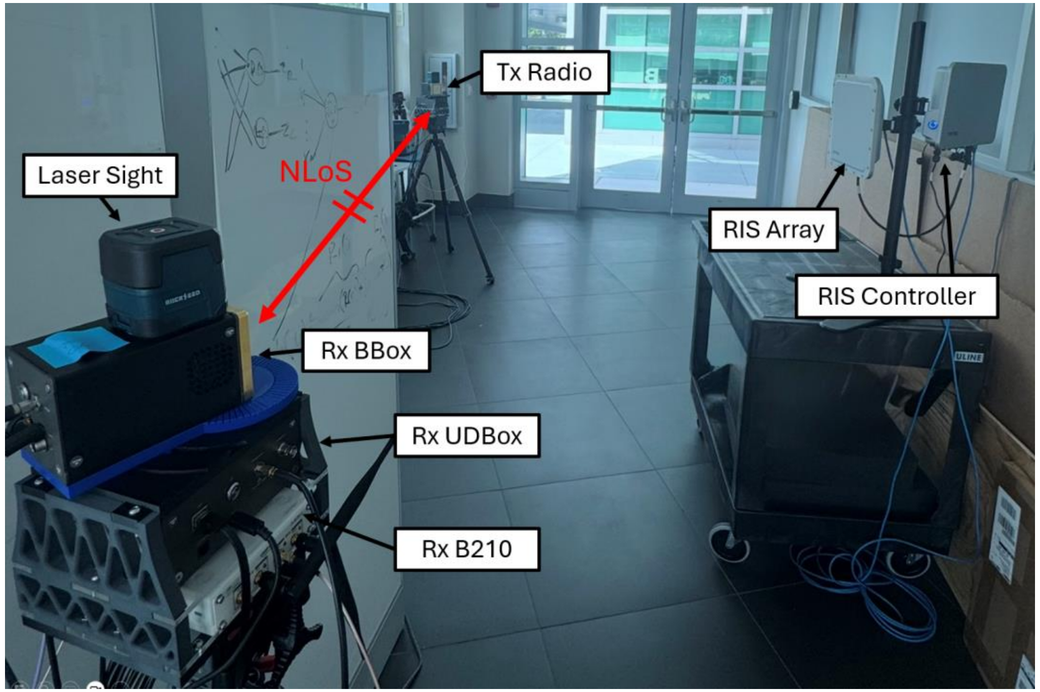
Problem

Assumptions

- Tx and Rx do not have a direct line-of-sight (LoS) path
- A reconfigurable intelligent surface (RIS) is strategically placed to provide an alternate virtual LoS path
- The true angles of arrival/departure between Tx, RIS and Rx are unknown

Problem: Given that Tx and Rx have no direct LoS and lack prior knowledge of relative geometry, how can we leverage a reconfigurable intelligent surface (RIS) — with controllable but discretized beam states — to dynamically establish and maintain a high-quality mmWave link?





Gap in Existing RIS Testbeds

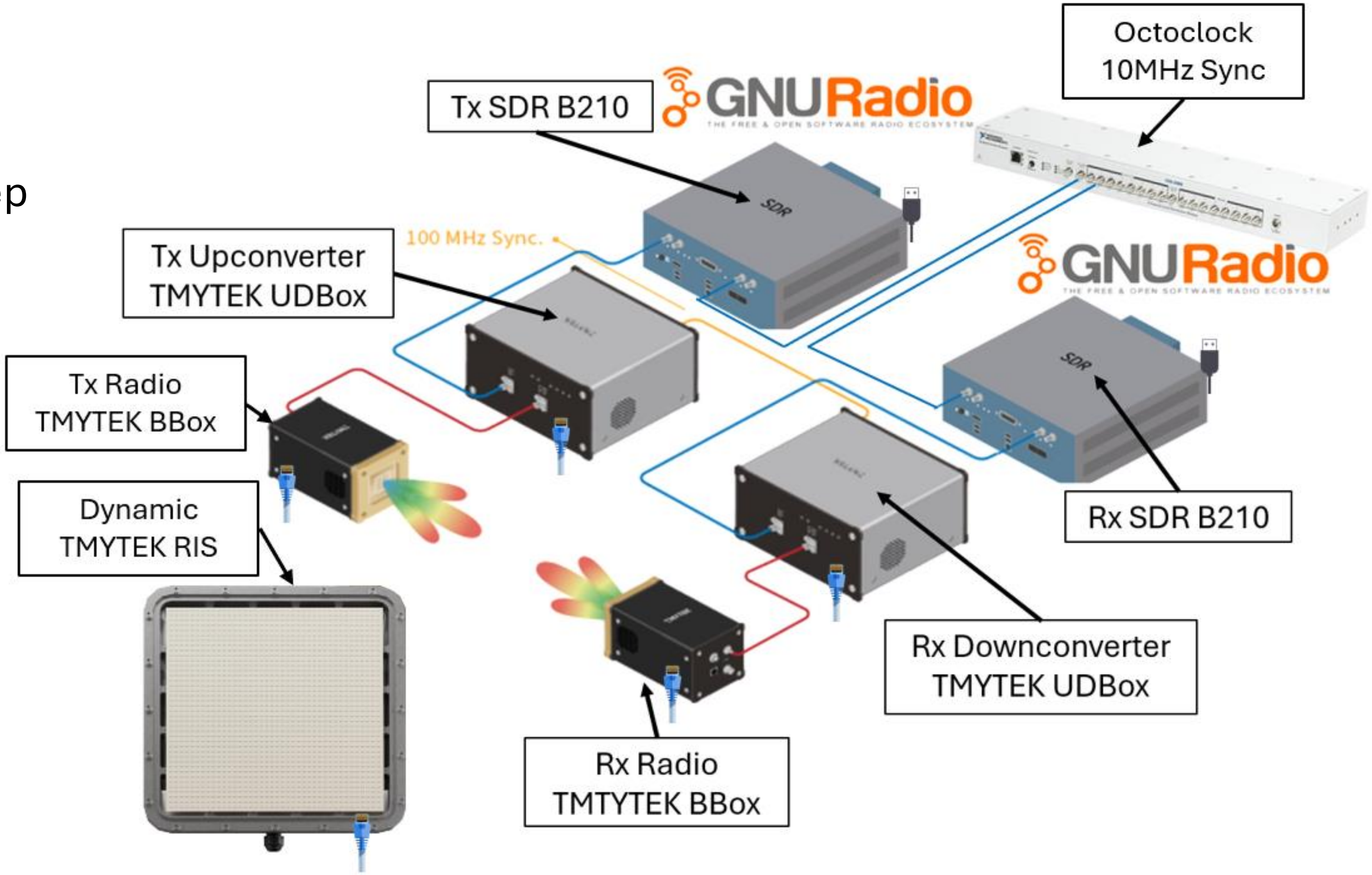
- **Sub-6 GHz RIS prototypes**
  - Show strong performance gains, but not representative of 28 GHz mmWave challenges
- **RIS hardware design innovations**
  - Focus on element-level control or exotic beam patterns, but lack tight SDR integration [2]
- **GNURadio orchestration efforts**
  - Demonstrate dynamic SDR–antenna control, but not extended to RIS at mmWave [3]

→ Missing: A unified, real-time testbed that integrates 28 GHz RIS hardware directly into the GNURadio loop, with latency and synchronization characterized

[2] Ouyang, et. al. Computer vision-aided reconfigurable intelligent surface-based beam tracking: Prototyping and experimental results. *IEEE Transactions on Wireless Communications*, 22(12):8681–8693, 2023.
[3] Agasti, et. al. Implementation of Software-Defined Antenna and Radio Test System for Congested Spectral Environments. *In Proceedings of the GNU Radio Conference*. 2023.

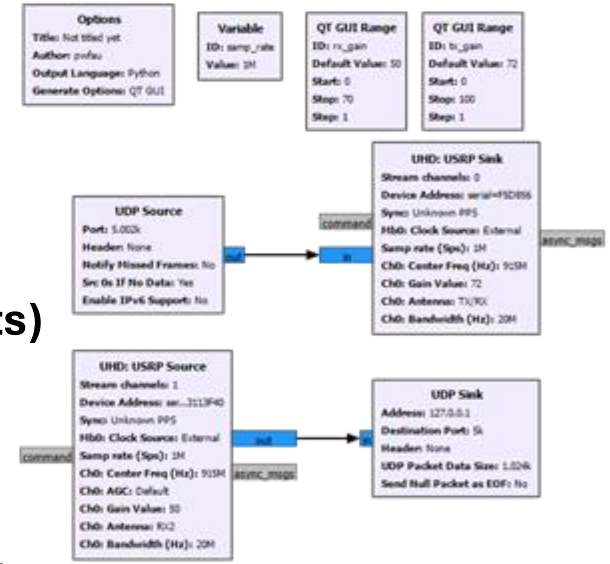
Hardware Setup

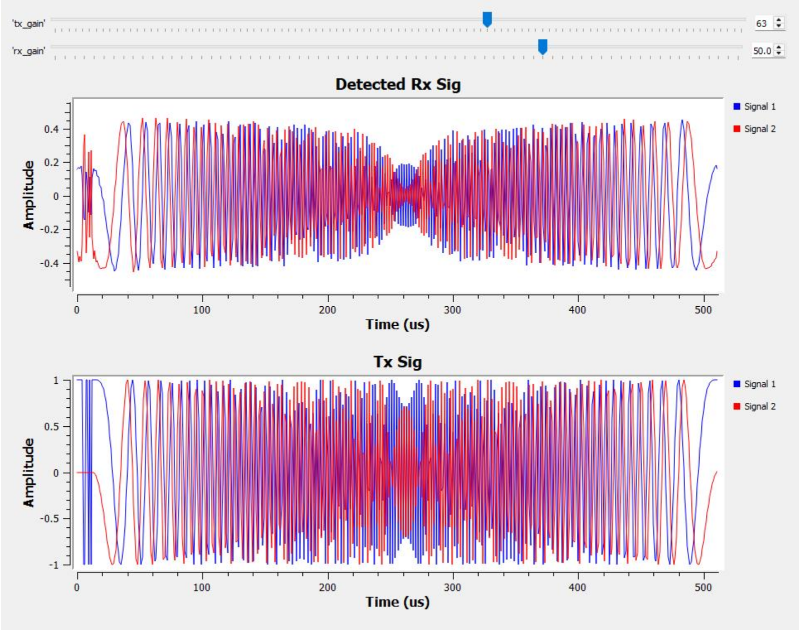
- **TMYTEK BBbox 5G**
  - 26.5 – 29.5 GHz
  - Antenna designed for 5G n257
  - Up to 16 controllable RF channels
  - Each channel: 360° phase shifter coverage with 5° per step
  - 15 dB attenuation range with 0.5 dB per step
- **TMYTEK UDBox 5G ultra-broadband**
  - NR mmWave frequency converter
  - IF: 0.01 – 14 GHz
  - Built-in LO with 24 – 44 GHz control range
- **TMYTEK XRifle Dynamic RIS**
  - 26 – 30 GHz
  - ± 60° range for both incident and reflection
- **USRP B210s**
- **GNURadio and python for baseband signal processing**



Software Framework and GNU Radio Integration

- **GNURadio Tx/Rx Flowgraph**
  - Initializes SDR hardware (Tx/Rx)
  - Handles IQ sample flow from/to USRPs
- **Beamsteering Interface (Python, UDP sockets)**
  - Controls TMYTEK Tx/Rx and RIS beams
  - Runs continuously in background
  - ~6 ms delay before beam changes take effect
- **Python Control Program (Master)**
  - Orchestrates data collection, data processing & logic for beamsteering
  - Generates Tx signals (modulation, preamble, etc.)
  - Implements Rx processing (synchronization, RSS calculation etc.)
  - Instructs both GNURadio and beamsteering interface





Experiment Design

Goal: Maximize received signal strength (RSS)

- **Setup assumptions:**
  - Tx ↔ Rx have no direct LoS
  - RIS provides alternative path
  - Elevation fixed → only azimuth angles matter
- **Parameters to optimize:**
  - RIS angle-of-incidence
  - RIS angle-of-reflection
  - RIS distance to Tx
  - RIS distance to Rx
- **Search strategy:**
  - Coarse-to-fine scanning
  - Iteratively refine best angles/distances
  - Inspired by prior RIS optimization work [4], adapted for TMYTEK hardware

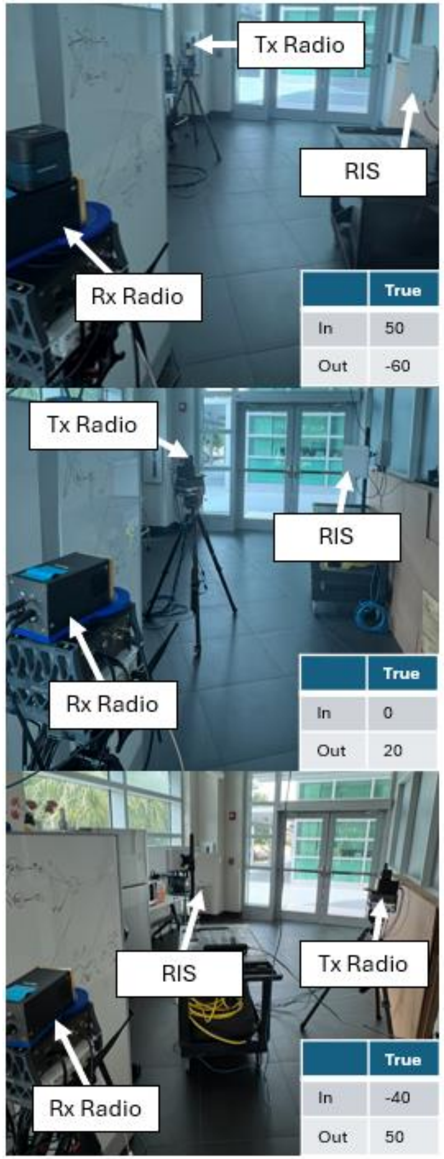
Algorithm 1 RIS Optimization

Initialize Tx and Rx beams at $0°$
Exclude elevation parameter (set to 0)
Initial search ranges:

$$\theta_{in}, \theta_{out} \in [-60°, 60°], \quad d_{in}, d_{out} \in [0, 3]\ m$$

Final target precision: $\Delta\theta_{\min} = 5°,\ \Delta d_{\min} = 0.2\ m$
Initialize coarse step sizes: $\Delta\theta = 20°,\ \Delta d = 1\ m$
while $\Delta\theta > \Delta\theta_{\min}$ **or** $\Delta d > \Delta d_{\min}$ **do**
  for $\theta_{in} = \theta_{in}^{rangestart}$ to $\theta_{in}^{rangeend}$ step $\Delta\theta$ **do**
    for $\theta_{out} = \theta_{out}^{rangestart}$ to $\theta_{out}^{rangeend}$ step $\Delta\theta$ **do**
      Measure RSSI for $(\theta_{in}, \theta_{out})$
    end for
  end for
  Select $(\theta_{in}^*, \theta_{out}^*)$ that maximize RSSI
  for $d_{in} = d_{in}^{rangestart}$ to $d_{in}^{rangeend}$ step Δd **do**
    for $d_{out} = d_{out}^{rangestart}$ to $d_{out}^{rangeend}$ step Δd **do**
      Measure RSSI for (d_{in}, d_{out})
    end for
  end for
  Select (d_{in}^*, d_{out}^*) that maximize RSSI
  Update search ranges to a neighborhood around $(\theta_{in}^*, \theta_{out}^*, d_{in}^*, d_{out}^*)$
  Refine step sizes: $\Delta\theta \leftarrow \Delta\theta/2$, $\Delta d \leftarrow \Delta d/2$
end while
Output:

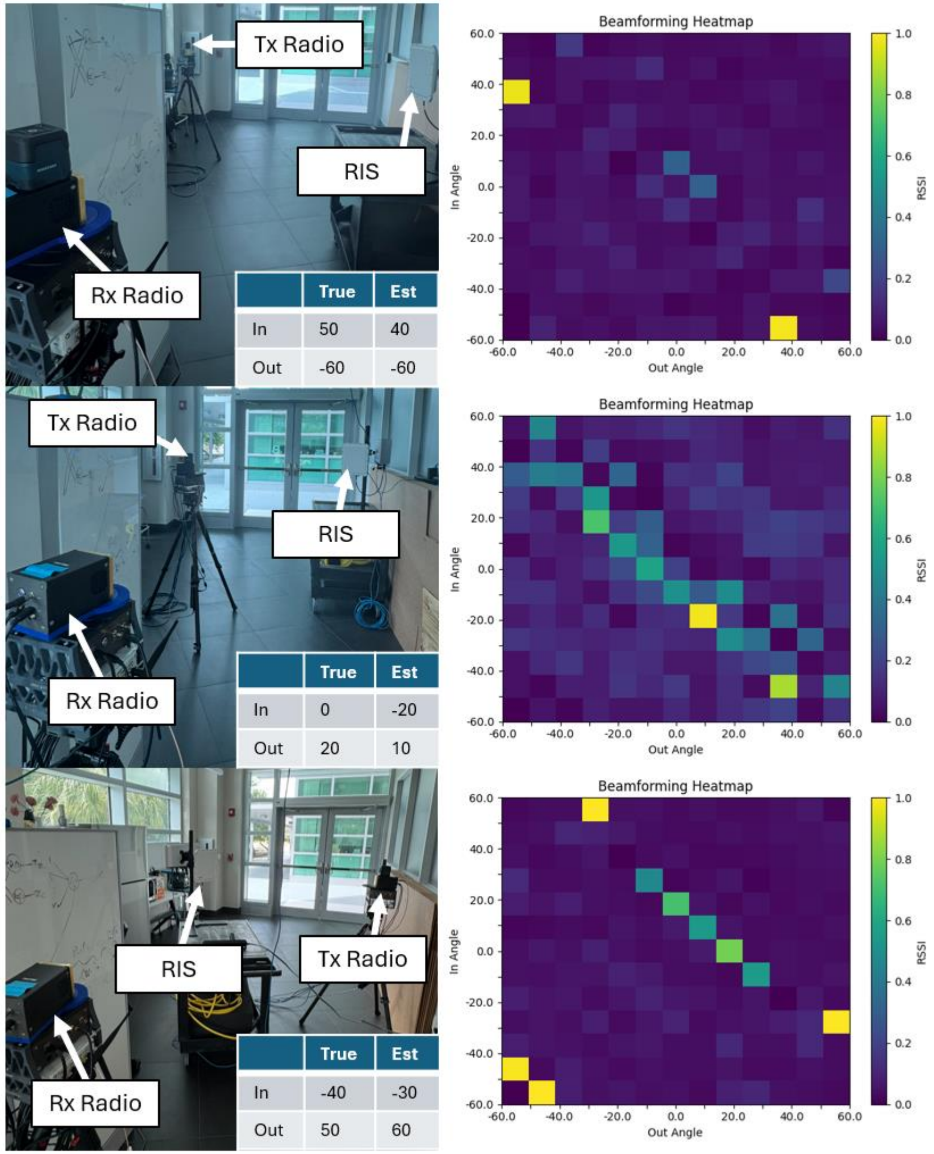
RSS-optimal RIS configuration: $(\theta_{in}^*, \theta_{out}^*, d_{in}^*, d_{out}^*)$



[4] Moghaddam, Shahriar and Moghaddam, Kiaksar. A group-based coarse-fine algorithm for intelligent reflecting surface beamforming. *Physical Communication*, 71:102668, 2025.

Experimental Results

- Optimal (in terms of max RSS) beam configuration found in 3 different environments
- Heatmaps show RSSI peaks near true inbound/outbound angles (or their negative counterpart)
- Symmetry effect: similar RSSI strength at ± angles due to RIS beam pattern
- RIS control loop consistently identified usable NLoS paths when LoS between Tx-Rx is blocked
- With RIS turned off, RSSI ≈ 0, confirming RIS role in NLoS scenarios
- GNU Radio provided flexibility in tuning SDR settings (e.g., Tx/Rx gains) in real-time for different environments



Challenges

- **Latency bottlenecks:**
  - Round-trip Tx→Rx delay: ~50-100 ms
  - Beam scan rate limited to ~10 Hz (too slow for mobile links)
- **Beam switching overhead:**
  - ~6 ms hardware settle time per beam change
  - Requires one-out-one-in transmissions → low duty cycle
- **System synchronization:**
  - Independent clocks across SDRs, RIS, and control PC
  - Must pause between transmissions to avoid angle ambiguity

Takeaway: Latency and synchronization constraints are the main barriers to scaling beam search to mobile, real-time scenarios

Conclusion and Future Work

- Developed a practical GNU Radio-RIS testbed at 28 GHz using off-the-shelf hardware
- Integrated Python backend for real-time RIS control and beam selection
- Demonstrated that system can maintain reliable mmWave links in NLoS conditions
- Showed optimal beam configurations can be discovered experimentally with minimal overhead
- Framework is reproducible and flexible, enabling future RIS research at higher frequencies

Future Work

- Reduce latency for faster beam switching (100s Hz target)
- Shift more logic into GNU Radio flowgraphs for efficiency
- Explore application-specific experiments (channel estimation, modulated signals, BER)
- Keep open-source repository updated and documented

Thank you!

Questions?

Contact: pwilmoth2023@fau.edu

Source code: https://github.com/C2A2-at-Florida-Atlantic-University/tmytek

GRCon25
Everett, Washington

FLORIDA ATLANTIC UNIVERSITY
Center for Connected Autonomy and Artificial Intelligence
College of Engineering & Computer Science

NSF
CENTER FOR SMART STREETSCAPES