Advanced Electrical & Computer Engineering

University of Kennesaw

Department of Electrical and Computer Engineering

Project 1: MIMO-GPR Virtual Array Synthesis

Authors:
Student 1 (Hossein Molhem)
Student 2 (ID: your id number)

Date: January 3, 2026

Contents

1 Project Proposal: MIMO-GPR Virtual Array Synthesis

This project serves as the first milestone in bridging Geophysics and Data Science. It will be demonstrated how a dense ”virtual” array can be mathematically generated using a sparse physical antenna array. This technique is considered fundamental to modern high-resolution Ground-Penetrating Radar (GPR) and seismic imaging in 2026.

1. Project Title

Subsurface Resolution Enhancement via MIMO Virtual Array Synthesis

2. Objective

To develop a Python-based simulation that calculates the spatial positions of virtual elements in a MIMO (Multiple-Input Multiple-Output) radar system and visualizes the resulting aperture expansion. This allows for higher angular resolution without increasing the number of physical sensors.

3. Geophysical Context

In GPR, the ability to resolve two closely spaced objects (like two parallel utility pipes) depends on the size of the array’s aperture. Using a MIMO configuration, we create a Virtual Array in which each virtual element represents the effective phase center of a distinct Transmit-Receive (TX-RX) pair.

4. Technical Roadmap & Learning Goals

Phase A: Mathematical Modeling

- **Coordinate Geometry:** Define positions for a set of M transmitters and N receivers (e.g., a ”Minimum Redundancy Array” or a simple linear sparse layout).
- **Synthesis Algorithm:** Implement the vector addition of TX and RX coordinates. The virtual element position for the i-th transmitter and j-th receiver is typically calculated as:

$$POS_{virtual} = \frac{POS_{TX,i} + POS_{RX,j}}{2}$$

- **Redundancy Analysis:** Identify and address overlapping virtual elements that may arise when physical spacing is not optimized.

Phase B: Data Science Integration (Pandas & Seaborn)

- Data Structuring: Store TX, RX, and Virtual positions in a Pandas DataFrame. Use categorical labeling to distinguish antenna types.
- Seaborn Visualization:
  - Use `sns.scatterplot()` to plot the 1D or 2D array layout.
  - Map the `hue` and `style` parameters to the **"Antenna Type"** to clearly visualize the "physical vs. virtual" relationship.
  - Create a **"Density Plot"** of the virtual elements to identify **"effective aperture"** and gaps in the array.

Phase C: Evaluation Metrics

- **Aperture Gain:** Calculate the ratio of the virtual array length to the physical array length.
- **Resolution Prediction:** Estimate the theoretical angular resolution improvement using the formula $\theta \approx \lambda/D$, where D is the new virtual aperture size.

5. Expected Outcomes

- A reusable Python script capable of testing different **MIMO geometries** (e.g., cross-arrays, box arrays, or random sparse arrays).
- High-quality **Seaborn plots** for your portfolio that demonstrate a deep understanding of geophysical signal processing.
- A foundational dataset of virtual positions that will be used in Project 2 (Subsurface Imaging).

6. Required Tools (2026 Stack)

- Python 3.12+
- **Pandas:** For managing complex multi-static pairings.
- **Seaborn/Matplotlib:** For high-fidelity geophysical visualization.
- **NumPy:** For vectorized spatial calculations.

2 Mathematical Foundation of MIMO-GPR Virtual Array Synthesis

2.1 Core Physical Principle: The Phase Center Theorem

The fundamental concept underpinning virtual array synthesis is the **Effective Phase Center** principle. For a bistatic radar system with transmitter at position $\mathbf{p}_t$ and receiver at position $\mathbf{p}_r$, the round-trip phase to a point target at position $\mathbf{p}_\tau$ is:

$$\phi = \frac{2\pi}{\lambda}\left(\|\mathbf{p}_t - \mathbf{p}_\tau\| + \|\mathbf{p}_r - \mathbf{p}_\tau\|\right) \tag{1}$$

For targets in the **far-field** (Fraunhofer region), where $\|\mathbf{p}_\tau\| \gg \|\mathbf{p}_t\|, \|\mathbf{p}_r\|$, we can approximate:

$$\phi \approx \frac{4\pi}{\lambda}\left\|\frac{\mathbf{p}_t + \mathbf{p}_r}{2} - \mathbf{p}_\tau\right\| \tag{2}$$

This demonstrates that the bistatic pair behaves as a **monostatic radar** located at the midpoint:

$$\mathbf{p}_v = \frac{\mathbf{p}_t + \mathbf{p}_r}{2} \tag{3}$$

3 Array Geometry and Virtual Element Generation

3.1 Formal Definition of Array Configuration

Let:

- M transmitters at positions: $\mathbf{P}_T = \{\mathbf{p}_{T,1}, \mathbf{p}_{T,2}, \ldots, \mathbf{p}_{T,M}\}$
- N receivers at positions: $\mathbf{P}_R = \{\mathbf{p}_{R,1}, \mathbf{p}_{R,2}, \ldots, \mathbf{p}_{R,N}\}$

The virtual array $\mathbf{P}_V$ contains $M \times N$ elements:

$$\mathbf{P}_V = \left\{\mathbf{p}_{V,ij} = \frac{\mathbf{p}_{T,i} + \mathbf{p}_{R,j}}{2} : i = 1, \ldots, M;\ j = 1, \ldots, N\right\} \tag{4}$$

4 Array Geometry and Virtual Element Generation

4.1 Linear Array Case (1D)

For a linear array along the x-axis:

- Transmitters: $\mathbf{p}_{T,i} = (x_{T,i},\ 0)$

- Receivers: $\mathbf{p}_{R,j} = (x_{R,j},\ 0)$

The virtual elements become:

$$x_{V,ij} = \frac{x_{T,i} + x_{R,j}}{2}$$

The aperture expansion ratio is:

$$R = \frac{\max(x_V) - \min(x_V)}{\max\left(\max(x_T),\ \max(x_R)\right) - \min\left(\min(x_T),\ \min(x_R)\right)} \tag{5}$$

5 Signal Model and Array Response

5.1 Received Signal Model

The received signal at receiver j from transmitter i for a point target at $\mathbf{p}_t$ is:

$$s_{ij}(t) = \alpha \cdot \exp\left(-j\frac{2\pi}{\lambda}(r_{T,i} + r_{R,j})\right) \cdot \mathrm{rect}\left(\frac{t - \tau_{ij}}{T_p}\right) \tag{6}$$

where:

$$\begin{aligned} r_{T,i} &= \|\mathbf{p}_{T,i} - \mathbf{p}_t\| \\ r_{R,j} &= \|\mathbf{p}_{R,j} - \mathbf{p}_t\| \\ \tau_{ij} &= \frac{r_{T,i} + r_{R,j}}{c} \\ \alpha &= \text{complex reflectivity} \end{aligned}$$

6 Signal Model and Array Response

6.1 Far-Field Approximation and Steering Vector

For a target at angle θ (broadside direction) in the far-field:

$$s_{ij}(\theta) = \alpha \cdot \exp\left(-j\frac{2\pi}{\lambda}(x_{T,i} + x_{R,j})\sin\theta\right) \tag{7}$$

This can be rewritten using the virtual element position:

$$s_{ij}(\theta) = \alpha \cdot \exp\left(-j\frac{4\pi}{\lambda}x_{V,ij}\sin\theta\right) \tag{8}$$

The array steering vector for the virtual array is:

$$\mathbf{a}_V(\theta) = \left[\exp\left(-j\frac{4\pi}{\lambda}x_{V,1}\sin\theta\right),\ \ldots,\ \exp\left(-j\frac{4\pi}{\lambda}x_{V,MN}\sin\theta\right)\right]^T$$

7 Redundancy Metric

Define the redundancy matrix $\mathbf{R}$ with elements:

$$R_{ij,kl} = \delta(x_{V,ij} - x_{V,kl}) \tag{9}$$

where $\delta(\cdot)$ is the Kronecker delta. The redundancy count for virtual position x is:

$$r(x) = \sum_{i=1}^{M} \sum_{j=1}^{N} \sum_{k=1}^{M} \sum_{l=1}^{N} \delta(x_{V,ij} - x) \cdot \delta(x_{V,kl} - x) \tag{10}$$

8 Minimum Redundancy Arrays (MRA)

For M transmitters and N receivers, the optimal physical spacing minimizes:

$$J = \sum_{x \in \mathcal{X}_V} (r(x) - 1)^2 \tag{11}$$

subject to constraints on total array length.

9 Resolution Analysis

9.1 Rayleigh Resolution Criterion

The angular resolution for a uniform linear array of length D is:

$$\Delta\theta = \frac{\lambda}{2D\cos\theta_0} \quad \text{(for monostatic array)} \tag{12}$$

For the virtual array with aperture D_V:

$$\Delta\theta_V = \frac{\lambda}{2D_V\cos\theta_0} \tag{13}$$

The resolution improvement factor is:

$$F = \frac{\Delta\theta}{\Delta\theta_V} = \frac{D_V}{D} \tag{14}$$

9.2 Beamwidth Calculation

The array factor for the virtual array is:

$$AF(\theta) = \sum_{m=1}^{M} \sum_{n=1}^{N} w_{mn} \exp\left(-j\frac{4\pi}{\lambda} x_{V,mn} \sin\theta\right) \tag{15}$$

The half-power beamwidth is approximately:

$$\theta_{BW} \approx 0.886 \frac{\lambda}{2D_V \cos\theta_0} \quad \text{[radians]} \tag{16}$$

10 Statistical Analysis of Virtual Array Properties

10.1 Virtual Element Density

The probability density function of virtual element positions for randomly placed physical elements follows:

$$f_X(x) = (f_T * f_R)(2x) \tag{17}$$

where f_T and f_R are PDFs of transmitter and receiver positions, and $*$ denotes convolution.

10.2 Aperture Gain Distribution

For a random MIMO configuration, the virtual aperture gain follows:

$$G = \frac{\max(X_V) - \min(X_V)}{\max(\max(X_T), \max(X_R)) - \min(\min(X_T), \min(X_R))} \tag{18}$$

with expected value:

$$\mathbb{E}[G] = 1 + \frac{\sigma_T^2 + \sigma_R^2}{(\mu_T - \mu_R)^2 + \sigma_T^2 + \sigma_R^2} \tag{19}$$

11 Extension to 2D Arrays

For 2D arrays with coordinates (x, y), the virtual array position vector is defined as:

$$\mathbf{p}_{V,ij} = \left(\frac{x_{T,i} + x_{R,j}}{2}, \frac{y_{T,i} + y_{R,j}}{2}\right) \tag{20}$$

The point spread function (PSF) in the spatial frequency domain (k_x, k_y) is:

$$PSF(k_x, k_y) = \left|\sum_{i=1}^{M} \sum_{j=1}^{N} \exp\left(-j(k_x x_{V,ij} + k_y y_{V,ij})\right)\right|^2 \tag{21}$$

12 Implementation Considerations

12.1 Computational Complexity

The virtual array generation has complexity $O(MN)$, while the beamforming has complexity $O(MN \times N_\theta)$, where N_θ is the number of angle bins.

12.2 Numerical Precision

The phase calculation requires careful handling of numerical precision:

$$\phi_{ij} = \text{mod}\left(\frac{4\pi}{\lambda} x_{V,ij} \sin\theta, 2\pi\right) \tag{22}$$

with modulo operation to avoid phase wrapping errors.
Intelligent spatial diversity in transmission and reception enables synthesis of a virtual array with significantly larger effective aperture than the physical array, enhancing resolution in subsurface imaging applications.

References