Dual-Beam DFRC Precoding for Joint Communications & Sensing

Haytam Zerdani

14/09/2025

Abstract

This report presents a minimal, reproducible demonstrator of dual-function radar-communications (DFRC) beamforming on a uniform linear array (ULA). We co-design a transmit weight vector to maintain downlink communications SINR towards a user equipment (UE) while meeting a sensing mainlobe constraint towards a look-angle and suppressing sidelobes elsewhere. A ridge-regularized pattern-fitting approach is used to approximate constrained designs and to sweep sensing gain targets, yielding a Pareto curve of "UE SINR vs sensing gain". The code is lightweight (Python) and intended as an entry point for 6G ISAC studies.

1 Objectives

- Form a UE beam and a sensing beam with one digital precoder on a ULA.

- Quantify the trade-off between communications SINR and sensing mainlobe gain.

- Enforce sidelobe suppression over a broad angular sector.

- Provide a compact, reproducible codebase and figures.

2 System Model

Consider an M-element ULA with inter-element spacing $d = \lambda/2$. The steering vector at angle θ (broadside at 0) is

$$\mathbf{a}(\theta) = \begin{bmatrix} 1 & e^{j2\pi\frac{d}{\lambda}\sin\theta} & \ldots & e^{j2\pi\frac{d}{\lambda}(M-1)\sin\theta} \end{bmatrix}^\top. \tag{1}$$

Let $\mathbf{w} \in \mathbb{C}^M$ be the transmit weights. The array response towards θ is $B(\theta) = \mathbf{a}(\theta)^H \mathbf{w}$ and the normalized power pattern is $|B(\theta)|^2$.

The UE effective channel is modeled as

$$\mathbf{h}_{\mathrm{UE}} = \frac{\mathbf{a}(\theta_{\mathrm{UE}}) + \kappa\,\mathbf{h}_{\mathrm{NLoS}}}{\|\mathbf{a}(\theta_{\mathrm{UE}}) + \kappa\,\mathbf{h}_{\mathrm{NLoS}}\|_2}, \tag{2}$$

with small Rayleigh perturbation $\mathbf{h}_{\mathrm{NLoS}}$ and $\kappa \ll 1$. The single-user downlink SINR (noise variance σ^2) is

$$\mathrm{SINR}_{\mathrm{UE}} = \frac{|\mathbf{h}_{\mathrm{UE}}^H \mathbf{w}|^2}{\sigma^2}. \tag{3}$$

3 Design Formulation

The idealized constrained problem is

$$\max_{\mathbf{w}} \quad |\mathbf{h}_{\mathrm{UE}}^H \mathbf{w}|^2 \tag{4}$$

$$\text{s.t.} \quad |\mathbf{a}(\theta_s)^H \mathbf{w}|^2 \geq G_{\min}, \tag{5}$$

$$|\mathbf{a}(\theta)^H \mathbf{w}|^2 \leq S_{\max}, \ \forall \theta \in \Theta_{\mathrm{SL}}, \tag{6}$$

$$\|\mathbf{w}\|_2^2 \leq P_{\max}.$$

To obtain a simple, robust baseline and sweep the trade-off efficiently, we use *ridge-regularized pattern fitting*: enforce complex gains at a small set of target angles and penalize sidelobes elsewhere,

$$\min_{\mathbf{w}} \ \|\mathbf{A}\mathbf{w} - \mathbf{b}\|_2^2 + \lambda \|\mathbf{w}\|_2^2, \qquad \Rightarrow \ \mathbf{w} = (\mathbf{A}^H \mathbf{A} + \lambda \mathbf{I})^{-1} \mathbf{A}^H \mathbf{b}. \tag{7}$$

Rows of $\mathbf{A}$ stack $\mathbf{a}(\theta_{\mathrm{UE}})$, $\mathbf{a}(\theta_s)$ with desired complex gains $b_{\mathrm{UE}} = 1$ and $b_s = \sqrt{G_{\mathrm{targ}}}$, and a dense set of sidelobe angles with target 0 weighted by w_{SL}. The solution is normalized to meet a transmit power constraint.

4   Methodology

Angles and Grid.   Choose θ_{UE} and sensing angle θ_s. Build a grid Θ_{SL} covering $[-90°, 90°]$ while excluding small windows around the targets.

Sweep.   For gain targets G_{targ} (in dB on voltage, i.e., $20 \log_{10}$), solve the ridge system and compute the pair $(G_{\mathrm{targ}}, \mathrm{SINR}_{\mathrm{UE}})$ to form a Pareto curve.

Robustness. λ regularizes against ill-conditioning and phase errors. w_{SL} controls sidelobe suppression.

5   Implementation

- `precoding/ula.py`: steering vector, beampattern, UE channel generator.

- `precoding/pattern_fit.py`: ridge design and SINR utility.

- `sims/dfrc_beam.py`: end-to-end sweep producing figures.

Run:

```
python - <<'PY'
import sys
sys.path.append('/mnt/data/isac')
from sims.dfrc_beam import run
run() # saves figures in data/results/
PY
```

6   Results

Figure 1 shows the normalized beampattern for a mid operating point (e.g., $M{=}12$, $\theta_{\mathrm{UE}}{=}10°$, $\theta_s{=}40°$). A mainlobe is steered to θ_s while the UE direction retains a strong response and sidelobes are suppressed across the remaining sector.

Figure 2 reports the Pareto frontier between sensing mainlobe *target voltage gain* and achieved UE SINR. As the sensing constraint tightens, the UE SINR decreases, quantifying the DFRC trade-off on a single array.
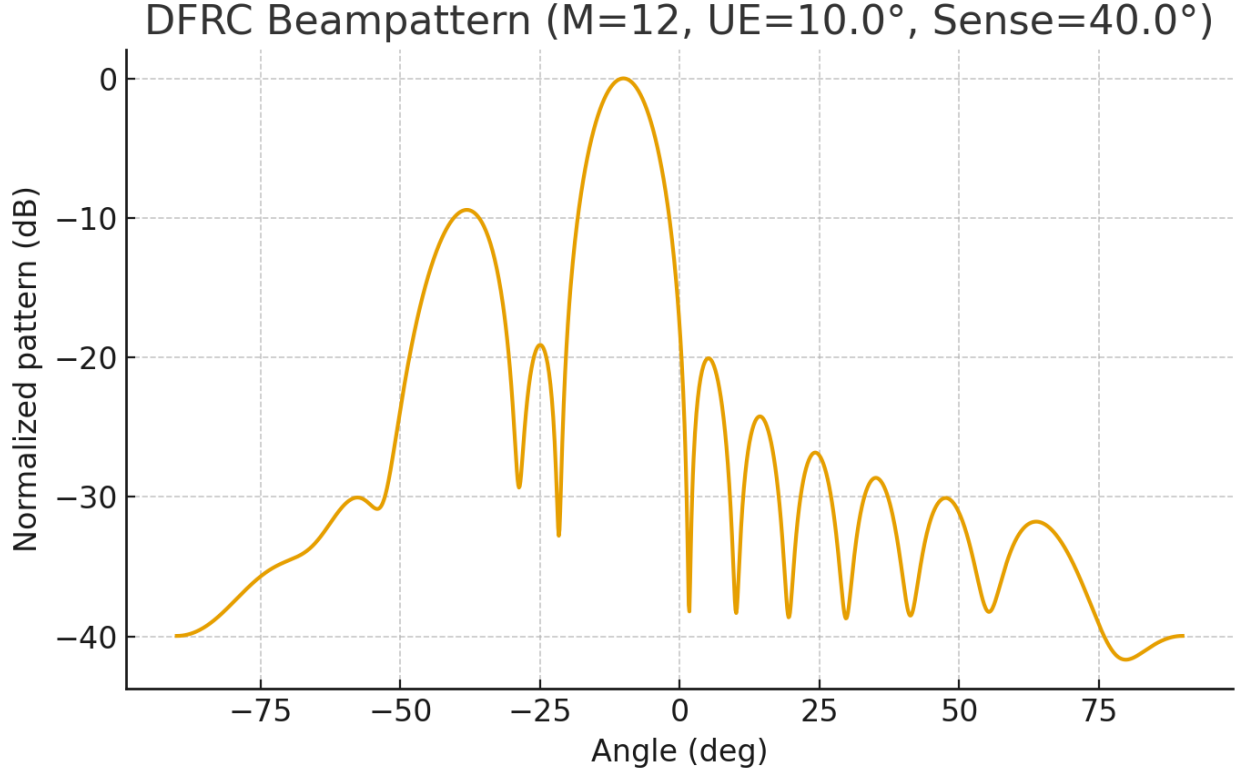
Figure 1: Normalized beampattern for a representative design.

7 Key Parameters

Parameter	Default
Elements M	12
Spacing d/λ	0.5
UE angle θ_{UE}	$10°$
Sensing angle θ_s	$40°$
Sidelobe mask exclude window	$\pm 8°$ around targets
Ridge weight λ	10^{-3}
Sidelobe weight w_{SL}	8
Noise variance σ^2	1
Gain sweep (voltage, dB)	$\{-10, -7, -5, -3, 0, 3, 6\}$

8 Discussion

When ridge is enough. For single-UE and one sensing lobe, ridge fitting with dense sidelobe samples gives clean, stable patterns and an interpretable Pareto. It is fast and differentiable.

When to upgrade. If hard inequality constraints are vital (e.g., guaranteed sidelobe caps), upgrade to QP/SOCP with slack variables. For multi-UE or multi-look sensing, extend **A** and add per-direction priorities or iterative reweighting.

Hardware non-idealities. Phase quantization, mutual coupling, and calibration errors can be absorbed by: (i) adding perturbations in simulation, (ii) increasing λ, and (iii) re-fitting with measured array responses.
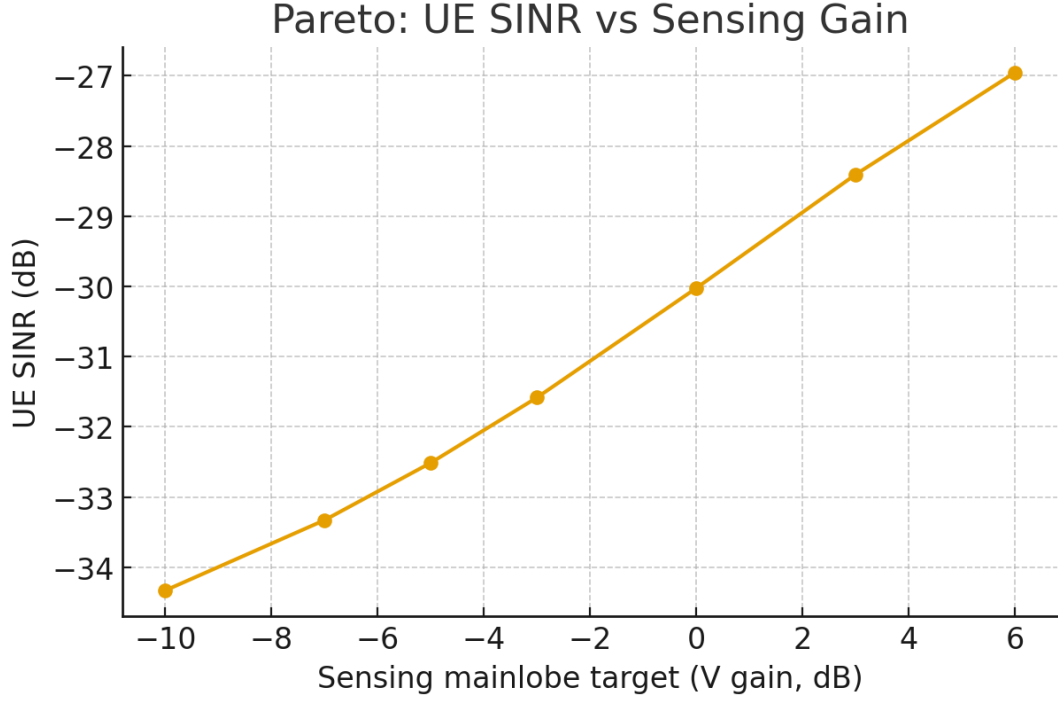
Figure 2: Pareto curve: UE SINR vs sensing mainlobe target (voltage gain, dB).

9 Planned Extensions

- Multi-UE DFRC with interference constraints and weighted sum-rate objective.

- Robust design under phase errors and per-antenna power constraints.

- Joint waveform–beam co-design with sensing PRB scheduling.

- Integration with Project 1 to evaluate end-to-end Pd/BLER trade-offs.

10 Conclusions

The DFRC beamforming demo provides a compact method to co-steer communications and sensing with a single array. The ridge approach exposes a clear SINR–sensing trade-off and serves as a fast baseline before moving to constrained convex formulations and hardware-in-the-loop tests.

Repo: `/mnt/data/isac`   **Figures:** `dfrc_beampattern.png`, `dfrc_pareto.png`