Design of Large Effective Apertures for Millimeter Wave Systems using a Sparse Array of Subarrays

Anant Gupta, *Student Member, IEEE*, Upamanyu Madhow, *Fellow, IEEE* Amin Arbabian, *Senior Member, IEEE* and Ali Sadri

***Abstract*—We investigate synthesis of a large effective aperture using a sparse array of subarrays. We employ a multi-objective optimization framework for placement of subarrays within a prescribed area dictated by form factor constraints, trading off the smaller beam width obtained by spacing out the subarrays against the grating and side lobes created by sparse placement. We assess the performance of our designs for the fundamental problem of bearing estimation for one or more sources, comparing performance against estimation-theoretic bounds. Our tiled architecture is motivated by recent progress in low-cost hardware realizations of moderately sized antenna arrays (which play the role of subarrays) in the millimeter wave band, and our numerical examples are based on 16-element (4×4) subarrays in the 60 GHz unlicensed band.**

***Index Terms*—Millimeter Wave Radar, Estimation Bounds, Compressive Estimation, Gridless Super-Resolution, Sparse Subarray Design, Multi-objective Optimization.**

I. Introduction

Many sensing and situational awareness applications (e.g., radar imaging for vehicles and drones) require highly directional, electronically steerable beams. Reducing beam width requires expansion of antenna aperture. This is typically accomplished by filling the aperture with antenna elements spaced at half the carrier wavelength or less, in order to avoid grating lobes. However, this approach does not scale well with aperture size since the cost, power consumption and design complexity increases with number of antenna elements.

In this paper, we investigate the problem of synthesizing narrow beams using a tiled architecture, with a sparse set of subarrays spread over a large physical aperture. Each subarray is a relatively compact antenna array with a moderate number of elements at sub-wavelength spacing. This is a modular design, in which each subarray can be controlled by a radio frequency integrated circuit (RFIC) of moderate complexity, with multiple RFICs tiled to build up large aperture arrays with a much larger number of elements compared to that of a single subarray. The resulting array is "sparse" because, while the total number of antenna elements summed across subarrays is large, this number is far smaller than that for a classical design with antennas at sub-wavelength spacing spanning the entire physical aperture. The sidelobes and grating lobes resulting from such spatial undersampling must therefore be controlled in order for our proposed "array of subarrays" to be useful. Our goal here is to determine the placement of a given number of subarrays over a physical aperture in order to optimize multiple beam attributes, including beam width, maximum sidelobe level, and directivity.

While our framework is general, the design of millimeter wave arrays is of particular interest to us, because the small carrier wavelength enables synthesis of narrow beams using relatively compact apertures. As a running example throughout this paper, we consider the design of a 60 GHz array of subarrays created by placing 8 subarrays over an aperture size of 10 cm by 10 cm ($20\lambda \times 20\lambda$ for wavelength $\lambda = 0.5$ cm), where each subarray has 4×4 elements arranged in uniform rectangular grid with 0.5λ horizontal spacing and 0.6λ vertical spacing. The total number of antenna elements in such designs is 128, which is an order of magnitude smaller than the 1600 elements required to cover the entire aperture at half-wavelength spacing. In addition to optimizing beam characteristics, our design framework also accounts for practical placement constraints consistent with existing prototype subarrays. For example, the subarrays need to be aligned along their axes, assuming that all elements have unidirectional linear polarization. Also, each subarray tile occupies extra physical area on the plane, which must be accounted for in the placement procedure.

A. Contributions

Our contributions are summarized as follows:

• We formulate the problem of subarray placement as multi-objective optimization of key performance measures such as beam width (BW), maximum sidelobe level (MSLL), eccentricity (ecc) and directivity (G_D),

$$\begin{aligned} \text{Minimize} \quad & \texttt{BW}(\boldsymbol{C},\boldsymbol{w}), \texttt{MSLL}(\boldsymbol{C},\boldsymbol{w}), \texttt{ecc}(\boldsymbol{C},\boldsymbol{w}) \\ \text{Maximize} \quad & G_D(\boldsymbol{C},\boldsymbol{w}) \\ \text{subject to} \quad & \text{AoS}(\boldsymbol{C}) \end{aligned} \tag{1}$$

where $\boldsymbol{C}$ is $N_s \times 2$ Subarray center position matrix, $\boldsymbol{w}$ is $N \times 1$ beamsteering weight vector and $\text{AoS}(\boldsymbol{C})$ are physical constraints to avoid overlapping subarrays. Note that orientation of subarrays is not an optimization variable in this architecture, since the polarization of the elements has to be aligned for beamforming. We consider minimization of a weighted linear combination of the objectives, focusing mainly on beamwidth BW and maximum sidelobe level MSLL. The configuration $\boldsymbol{C}$ that we optimize over is characterized by a set of discrete-valued variables, and the number of possible values for these variables is combinatorially explosive. Furthermore, we do not have closed form expressions for the performance

A. Gupta and U. Madhow are with the Department of Electrical and Computer Engineering, University of California Santa Barbara, CA USA (Email: {anantgupta, madhow}@ece.ucsb.edu)

A. Arbabian is with the Department of Electrical Engineering, Stanford University, CA USA (Email: arbabian@stanford.edu)

A. Sadri is with Intel Corporation (Email: ali.s.sadri@intel.com

measures as a function of $\boldsymbol{C}$, hence significant computation is required to evaluate the cost function for each configuration. In order to control the complexity, we use geometric heuristics to eliminate similar configurations in the first stage of our algorithm, and then employ a second stage of refinement using small perturbations around the first stage solution. We numerically explore the Pareto front for (1) by sweeping through the weights, and illustrate how the beam attributes depend on the weights.

• We evaluate our designs using estimation-theoretic benchmarks for two-dimensional (2D) direction of arrival (DoA) estimation. At low signal-to-noise ratio (SNR), large sidelobes can lead to large errors in the DoA estimate. At high SNR, on the other hand, the DoA estimation error is governed by beam width. We derive a Ziv-Zakai bound (ZZB), which captures the effect of both large and small estimation errors, for DoA estimation for specular paths. The ZZB exhibits a distinct transition in its behavior from low to high SNR, tending at high SNR to the Cramer-Rao bound (CRB), which captures the effect of small errors around the true parameter value. Thus, we use the ZZB transition SNR as a measure of efficacy of sidelobe reduction, and the CRB as a measure of efficacy of beam width reduction.

• We report in detail on two array designs, $A1$ with primary emphasis on reducing beamwidth and $A2$ based on joint optimization of beamwidth and maximum sidelobe level. These designs are compared against two benchmark arrays. The first is termed a "compact array," with subarrays packed closely together: this is expected to have worse beam width but smaller sidelobes than our sparse designs. The second is termed a "naive array," obtained by placing subarrays in diamond pattern to obtain beamwidth equivalent to that of sparse array *A2*. Some illustrative numerical results for our running example are as follows. The sparse design *A1* is 11 dB better than the compact array in terms of CRB, while degrading less than 1 dB in terms of ZZB threshold. The sparse design *A2* is 4 dB better in terms of CRB than the "naive array," while also having a better ZZB threshold.

• We investigate DoA estimation performance numerically using a state of the art algorithm for off-grid estimation. The impact of the higher sidelobes due to sparse placement, and hence that of our optimization procedure, is more evident when estimating DoA in the presence of multiple interfering targets. We show that, depending on the strength of interferers, our optimized arrays achieve better estimation accuracy than the "compact" and "naive" benchmark arrays at moderate to high SNR due to a combination of sharper beamwidth and lower sidelobes. We also show that the efficacy of DoA estimation using our sparse designs, and the associated benchmarks, is maintained when we employ compressive measurements.

B. Related Work

There is a rich body of work on sparsifying linear arrays, including minimum redundancy arrays [1], genetic optimization [2], joint Cramér Rao Bound and sidelobe level optimization [3], and simulated annealing [4]. Most popular design strategies try to find an element pattern which minimizes beamwidth, along with some notion of DoA ambiguities such as sidelobe level or probability of DoA outlier. Recent approaches like Nested 2D arrays [5] and H-arrays [6] utilize the idea of "difference co-array" to reduce the number of redundant spacings and maximize the randomness of element positions, so that the number of spatial frequencies being sampled by the array is maximized.

A closely related sparse array design methodology is the *sensor selection* problem, wherein a smaller subset of individual antenna positions is to be chosen from a predefined grid. By employing certain surrogate measures, near-optimal arrays can be obtained in polynomial time using standard convex relaxation methods [7], [8]. Array thinning methods such as these are well known to avoid complicated nonlinear optimizations for linear case [9].

Most existing techniques, however, assume that antenna elements can be placed freely. Hence, they do not apply in our setting, where element placement within subarrays is constrained. The prior work most similar to our is [10], which investigates design of linear arrays with two and three subarrays. However, the focus there is on performance criteria for comparing a number of sensible designs in a far smaller design space, rather than searching over a large space of possibilities as we do here.

Our performance evaluation requires implementation of DoA estimation algorithms. Classical subspace-based algorithms such as MUSIC [11] and ESPRIT [12], as well as their extensions to arrays of subarrays such as [13]–[16], rely on regular array geometries for efficient computation. Recently developed super-resolution algorithms such as Basis Pursuit Denoising (BPDN) [17] and Newtonized Orthogonal Matching Pursuit (NOMP) [18] are both more general and have better performance. It is worth noting that [19] shows that, for large arrays, BPDN and other sparse estimation techniques with compressive measurements outperform subspace-based methods. In this paper, we employ NOMP in our numerical experiments, since we have found it to provide better performance than BPDN at lower complexity. We also show that the performance trends are unchanged under compressive measurements, consistent with recent general theory [20].

C. Outline

We first describe the beam attributes to be optimized while designing the sparse arrays and discuss constraints for the optimization in Section II. The geometric heuristics and design approach are described in detail in Section III. We then provide a brief review of estimation bounds for 2D bearing estimation and discuss their utility for analyzing the sparse arrays in Section IV. Numerical results are provided in Section V. These include exploration of the Pareto front, and comparison of example designs against benchmarks in terms of both beam characteristics and DoA estimation. We show in Section VI that the performance trends hold for compressive DoA estimation, and conclude in Section VII.

II. Sparse Subarray Design

We formulate the array design problem in terms of jointly optimizing multiple beam parameters that are expected to

affect DoA estimation performance.

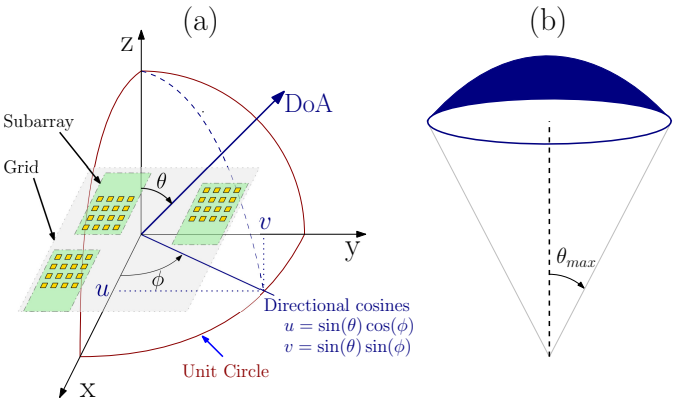


Fig. 1. (a) 2D Array Geometry and Spherical coordinate system. (b) ROI: Uniform distribution of 2D-DoA $\boldsymbol{u}$ in spherical cap with half angle θ_{max}

A. Beam Pattern Basics

We use the directional cosines

$$u = \sin(\theta)\cos(\phi), v = \sin(\theta)\sin(\phi) \tag{2}$$

to represent the DoA of target. The elevation θ and azimuth ϕ angles are measured from the broadside direction (perpendicular to the baseline array plane). The 2D beampattern $R(u,v)$ in direction (u,v) when the beam is steered towards the broadside is given by

$$R(u,v) = \frac{1}{N^2}\left|\sum_{i=1}^{N} e^{jk\left(ud_i^x + vd_i^y\right)}\right|^2 \tag{3}$$

where N is the number of array elements; $[d_i^x, d_i^y]^T \triangleq \boldsymbol{d}_i$ are the 2D co-ordinates of arrays elements and $k = \frac{2\pi}{\lambda}$ is the wavenumber. We assume isotropic antenna elements with ideal steering weights and far-field sources with normalized response. The term "subarray" refers to the subset of elements with uniform half-wavelength spacing, while "super-array" refers to the placement of these subarrays, which is described by the subarray centers. Since the elements in a subarray are fixed, the array element locations, $\boldsymbol{D}$ can be expressed in terms of the subarray centers, $\boldsymbol{C}$, as $\boldsymbol{D} = \boldsymbol{C}\otimes\mathbb{1}_{N_e} + \boldsymbol{D}_e\otimes\mathbb{1}_{N_s}$, where $\boldsymbol{D}_e$ is the fixed $2\times N_e$ matrix containing the subarray element coordinates with respect to its center, N_s is the number of subarrays, N_e is number of elements in individual subarrays, $\mathbb{1}_n$ is an $n\times 1$ column vector of ones, and $\otimes$ denotes the Kronecker product.

When beamforming in a general direction (u_0, v_0) (broadside corresponds to $(u_0, v_0) = (0,0)$), the beam pattern is given by

$$R_{(u_0,v_0)}(u,v) = R(u-u_0, v-v_0) \tag{4}$$

For Direction of Arrival (DoA) estimation, the ideal beam should have small beamwidth with minimal sidelobes and high directivity. We consider the following beam attributes, some of which depend on the steering direction (u_0, v_0), as key performance metrics to be optimized:

- *2D beamwidth* (BW): Although the main beam of non-uniform array has non-trivial shape in 2D, we approximate it as an ellipse to define beamwidth. We evaluate the 2D beamwidth in terms of the 3-dB beamwidths along the major and minor axes of this ellipse, denoted by $\mathrm{BW}_{\max}$ and $\mathrm{BW}_{\min}$, respectively. The mean squared error of DoA estimation depends on the sum of these beamwidths (see Appendix A), hence we define beamwidth as $\mathrm{BW}^{\mathrm{DoA}} = \sqrt{\mathrm{BW}_{\max}^2 + \mathrm{BW}_{\min}^2}$.
- *Maximum sidelobe level* (MSLL): is the relative level of the strongest sidelobe in the beampattern with respect to the main lobe, $\mathtt{MSLL} = 10\log\left(\mathrm{R}_{\max}/\mathrm{R}_{\mathrm{msl}}\right)$. Thus, $R_{\max}$ and R_{msl} are the largest and second largest magnitude local maximas of beampattern $R(u,v)$ given by

$$\begin{aligned} R_{\max}(u_0,v_0) &= \max_{u,v} R_{u_0,v_0}(u,v) \\ &= R_{u_0,v_0}(u_0,v_0) = R(0,0) \\ R_{\mathrm{msl}} &= \max_{u^*,v^*} R_{u_0,v_0}(u^*,v^*) \\ &\quad \text{s.t. } (u^*,v^*) \neq (u_0,v_0), \\ &\quad R_{u_0,v_0}(u^*,v^*) \geq R_{u_0,v_0}(D_\epsilon(u^*,v^*)) \end{aligned}$$

  where $D_\epsilon(u^*,v^*) = \{(u,v) : |u-u^*| < \epsilon, |v-v^*| < \epsilon\}$ denotes ϵ-neighborhood. Note that $R_{\max}$ does not depend on steering direction (u_0, v_0), but R_{msl} might.
- *Directivity* ($\mathtt{G_D}$): The directivity is the ratio of main lobe power to average power, $\mathtt{G_D} = 10\log\left(\frac{\mathrm{R}_{\max}}{\mathrm{R}_{\mathrm{avg}}}\right)$ The average power does not have a closed form expression for general planar arrays, and is evaluated in (u,v) domain by the integral [21]:

$$R_{avg} = \frac{2}{4\pi}\int_{-1}^{1}\int_{-\sqrt{1-v^2}}^{\sqrt{1-v^2}} \frac{R_{u_0,v_0}(u,v)}{\sqrt{1-u^2-v^2}} dudv$$

- *Eccentricity* (ecc): is a measure of the asymmetry of the main beam. We add this additional parameter to suppress the trivial linear placement solution, $\mathtt{ecc} = \sqrt{1-\left(\mathrm{BW}_{\min}/\mathrm{BW}_{\max}\right)^2}$

For non-uniform planar arrays, none of these beam parameters have a closed form expression [22], hence they must be computed numerically. In our simulations, we compute these beam attributes using a beampattern over a 512×512 grid in UV space as shown in Figure 2.

B. Problem Formulation

In order to develop geometric heuristics for optimization, we first analyze the effect of increasing aperture width, keeping the number of antenna elements fixed, for linear and planar arrays, with uniform and subarray-based architectures as shown in the rightmost section of Figure 3. In the plots of beam attributes in Figure 3, the dashed line represents the aperture width for half-wavelength inter-element spacing, when the uniform and subarray-based configurations match.

- The MSLL for the array of subarrays increases much faster than for a uniform configuration due to a grating lobe appearing close to main beam. This attribute is sensitive to the element distribution and behaves unpredictably for non-uniform arrays.

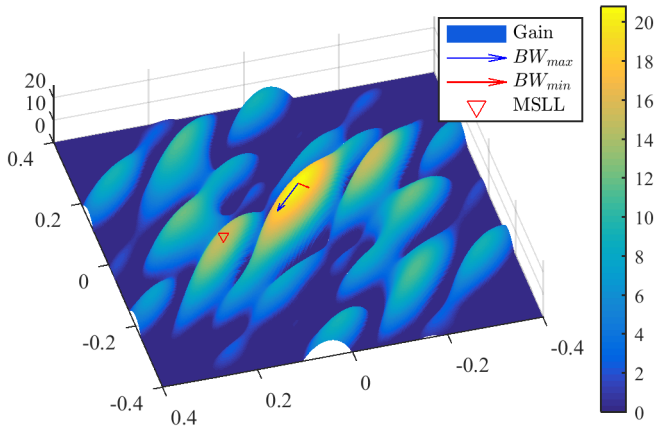


Fig. 2. Beam Attributes from array Beampattern.

- The 3dB Beamwidth ($\texttt{BW}^{max}$ for planar array) for both array types reduces congruently, confirming that it is inversely proportional to aperture width independent of the distribution of elements.
- Directivity increases as we increase the inter-element spacing, but only up to a certain limit, and then becomes constant [23]. This generalizes well to planar arrays as shown in the Figure 3. As one can see the directivity for subarrayed configurations remains approximately constant with increasing aperture width beyond standard spacing. We therefore do not include this metric in our cost function.

The objectives that we wish to trade off against each other do not have the same units: for example, MSLL is measured in dB relative to the maximum for the main lobe, whereas BW is measured in deg. We therefore normalize each raw objective value, o^{raw} by its range as follows:

$$o(\boldsymbol{C}) = \frac{o^{\text{raw}}(\boldsymbol{C}) - \min\limits_{\forall \boldsymbol{C}'} \{o^{\text{raw}}(\boldsymbol{C}')\}}{\max\limits_{\forall \boldsymbol{C}'} \{o^{\text{raw}}(\boldsymbol{C}')\} - \min\limits_{\forall \boldsymbol{C}'} \{o^{\text{raw}}(\boldsymbol{C}')\}}$$

The range of each objective is computed numerically while constructing the dictionary of all configurations, $\boldsymbol{C} \in \mathcal{C}$, described later in III-A.

The constrained multi-objective optimization can now be formulated as follows:

$$\begin{aligned} \boldsymbol{C}^*, \boldsymbol{w}^* = &\arg\min_{\forall \boldsymbol{C}, \boldsymbol{w}} \quad f(\boldsymbol{C}, \boldsymbol{w}) \\ &\text{subject to} \quad \text{AoS}(\boldsymbol{C}) \end{aligned} \tag{5}$$

where

$$f(\boldsymbol{C}, \boldsymbol{w}) = \alpha \texttt{BW}(\boldsymbol{C}, \boldsymbol{w}) + \beta \texttt{MSLL}(\boldsymbol{C}, \boldsymbol{w}) + \gamma \texttt{ecc}(\boldsymbol{C}, \boldsymbol{w}) \tag{6}$$

is the weighted cost function in terms of the normalized objective functions and $[\alpha, \beta, \gamma]$ are weights that can be used to sweep through the optimal surface for this optimization. We show some example arrays obtained for different choices of weights in Table I.

Since the cost function $f(\boldsymbol{C}, \boldsymbol{w})$ is evaluated numerically from its beampattern for specific values of $(\boldsymbol{C}, \boldsymbol{w})$, exploring the entire solution space is computationally infeasible. This discrete-valued nature of the optimization variables leads to a combinatorial problem without closed form objectives and constraints. Furthermore, beam characteristics in general depend on the steering weights $\boldsymbol{w}$, which in turn depend on the direction (u_0, v_0) in which we are steering. We therefore employ two key simplifications:

- We remove the dependence of the cost function on beamforming direction, and hence on steering weights, by computing the objectives based on an expanded beam pattern, as discussed in Section II-C.
- We employ geometric heuristics to cut down the solution space to a reasonable size, as described in Section III.

C. Invariance to Beamforming Direction

The cost function in (6) is evaluated using the beampattern $R(u-u_0, v-v_0)$, which depends on the beamsteering direction (u_0, v_0). It would be prohibitively expensive to evaluate the beam attributes over all such beampatterns for finding the optimal $(\boldsymbol{C}, \boldsymbol{w})$. However, for arrays with isotropic elements, we can define an Expanded Beam pattern (EBP) which subsumes beampatterns of all steering direction in a Region of Interest (ROI) [24]. Suppose that our maximum steering angle in the ROI is θ_{max}. From (2), we see that (u_0, v_0) lies within a circle of radius $\sin\theta_{max}$. On the other hand, sidelobes can appear at any (u, v) within a circle of radius 1. It is easy to see, therefore, that $(u - u_0, v - v_0)$ is guaranteed to lie within a circle of radius $1 + \sin\theta_{max}$. We can therefore compute beam attributes using the following EBP:

$$\begin{aligned} R_\rho(\tilde{u}, \tilde{v}) &= \frac{1}{N^2} \left| \sum_{i=1}^{N} e^{jk\rho\left(\tilde{u} d_i^x + \tilde{v} d_i^y\right)} \right|^2 \\ \rho &= 1 + \sin(\theta_{max}) \end{aligned} \tag{7}$$

Figure 4 shows the EBP, $R_{1.5}(\tilde{u}, \tilde{v})$ for ROI with $\theta_{max} = 30°$, and the beampattern for the steering angle ($(u_0, v_0) = (0.3, 0.4)$). The shape of the main beam is preserved under the transformation (7), hence beam width and eccentricity can be directly evaluated from (7). The MSLL evaluated from EBP is a worst-case value, corresponding to an argument $\rho(\tilde{u}, \tilde{v})$ which, in principle, might not correspond to a feasible value of $(u-u_0, v-v_0)$ in (4). However, the maximum sidelobe always lies within the main lobe of the subarray beam pattern, so that physically implausible values of $\rho(\tilde{u}, \tilde{v})$ do not correspond to large local maxima of the EBP.

With the introduction of the EBP, we can, without loss of generality, assume that the main beam is being steered towards broadside, setting $\boldsymbol{w} = \mathbb{1}_N$. Our problem now reduces to finding the optimal configuration $\boldsymbol{C}^*$ as follows,

$$\boldsymbol{C}^* = \arg\min_{\boldsymbol{C}} f(\boldsymbol{C}, \mathbb{1}_N) \quad \text{subject to} \quad AoS(\boldsymbol{C}) \tag{8}$$

III. Placement Optimization

In order to optimize the placement, we need to evaluate the cost function over all array configurations. The number of configurations depends on the allowed form factor, and the size of the subarray module, and an exhaustive search over all configurations is computationally infeasible: for example, the

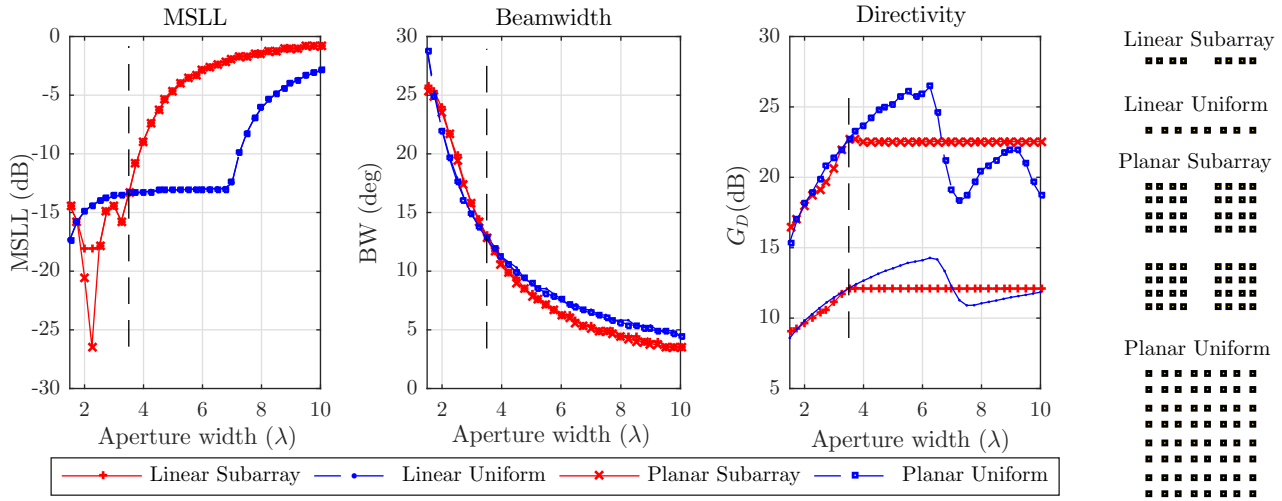


Fig. 3. Comparison of beam attributes of subarrayed and uniform architecture with increasing aperture width.

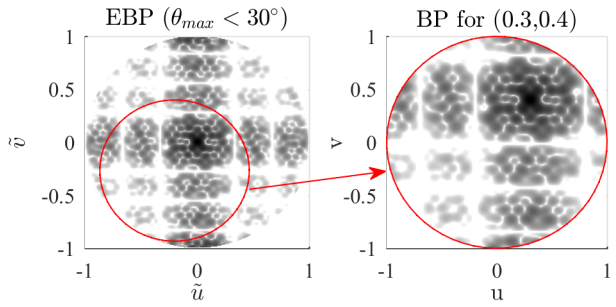


Fig. 4. Expanded Beampattern for ROI ($\theta_{max} < 30°$)

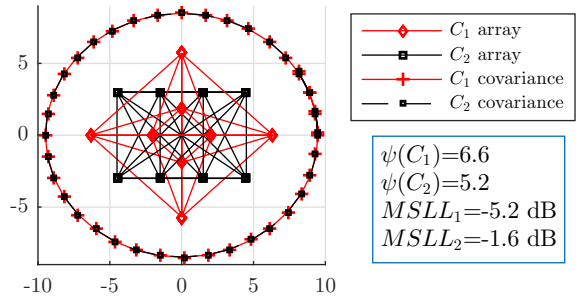


Fig. 5. Super-arrays with equal covariance but different beam attributes.

number of configurations for a discrete grid of size 20×20 is of the order 10^{20}. We therefore propose a two-stage approach, first performing a combinatorial search on a reduced search space, and then obtaining the final solution by searching over perturbations around the solution from the first stage.

A. Combinatorial search

We reduce the solution space by removing geometrically "similar" arrays. We employ the covariance of the element positions, and pairwise element separations, as measures of similarity. The choice of covariance of element positions $\mathbf{\Sigma}(\boldsymbol{C})$ as similarity metric is motivated by its inverse proportionality to Cramér Rao Bound on accuracy of DoA estimation (see Section IV-A1). However, array configurations with similar array covariance but diverse beam attributes also exist: Figure 5 shows an example of two array configurations with different shapes but the same covariance. The arrays have similar beamwidth but their MSLL levels are different. We observe that the variance of pairwise element distances $\psi(\boldsymbol{C})$ is different for these arrays, and use it as an indicator for these large scale deviations. This allows us to reduce the dimensionality of the solution space from $2 \times N_s$ down to 3 array shape parameters: the eigenvalues (λ_1, λ_2) of $\mathbf{\Sigma}(\boldsymbol{C})$ and $\psi(\boldsymbol{C})$.

1) Subarray Placement Algorithm: We construct a prefix tree dictionary to find feasible solutions using a breadth first search based enumeration technique. The element position covariance for an array of subarray can be uniquely represented by the covariance of its subarray centers, $\mathbf{\Sigma_D} = \mathbf{\Sigma_C} + \mathbf{\Sigma_{D_e}}$. Hence the super-array center covariance can be used instead of that of the full array in the Dictionary search algorithm. Each node in the tree stores a subarray center position, and the path from root to a node at the n^{th} layer of prefix tree represents a unique configuration of n subarrays. The subarray centers are constrained to lie on a fixed set of discrete grid points G. The

Algorithm 1 Prefix Tree Dictionary Search

1: INITIALIZE: $\mathcal{C}^1 = \left\{ \boldsymbol{C}_i^{(n=1)} \right\}; i \in [1, N_{init}]; n = 1$
2: **while** $n < N_s$ **do**
3: LIST all vacant Gridpoints $V_i = \mathcal{T}_G(\boldsymbol{C}_i^n); i \in [1, |\mathcal{C}^n|]$
4: APPEND subarray at vacancies V_i,
$\hat{\mathcal{C}}^{n+1} = \bigcup_{i=1}^{|\mathcal{C}^n|} \boldsymbol{C}_i^n \times V_i$
5: PRUNE: $\mathcal{C}^{n+1} \leftarrow$ Prune $\left(\hat{\mathcal{C}}^{n+1}\right)$
6: $n = n + 1$
7: **end while**
8: Return $\mathcal{C}^{N_s}$

algorithm is described in Algorithm 1. We briefly discuss the key steps below.

- INITIALIZE: In order to allow for sufficient exploration,

we employ multiple random initializations $\boldsymbol{C}_i^1$ of the roo node being placed on N_{init} different locations on the gri (For example, circular configurations cannot be obtaine if the root subarray is fixed at the center.)

- LIST: Define the operator $\Im_G : G^{|\boldsymbol{C}_i^n|} \rightarrow G^{|V_i|}$ whic maps the set of subarrays centers $\boldsymbol{C}_i^n \in \mathcal{C}^n$ to the s of $|V_i|$ vacant gridpoints in G available for placemer of next subarray which are not blocked by the subarray already placed at $\boldsymbol{C}_i^n$. This operator also accounts for additional surface area occupied by the subarray module apart from the physical antenna elements (see Appendix C for details).
- APPEND: The $(n+1)^{th}$ subarray configuration is constructed from the vacancies, $\boldsymbol{C}_i^n \times V_i$ where $\times$ denotes cartesian product of sets. A temporary dictionary $\hat{\mathcal{C}}^{n+1}$ is formed by inserting $|V_i| = \kappa$ child nodes for each node in the n^{th} layer.
- PRUNE: Nodes corresponding to "similar" configurations are deleted based on the array shape parameters
  1) Find eigenvalues (λ_1, λ_2) of the subarray center covariance matrix, $\Sigma(\boldsymbol{C})$ and variance of array separations, $\psi(\boldsymbol{C}) = \mathbb{E}\left[(l_{ij} - \mathbb{E}[l_{ij}])^2\right]$, where, l_{ij} denotes the distances betweeen i^{th} and j^{th} elements.
  2) Enumerate unique configurations by binning the $(\lambda_1, \lambda_2, \psi)$ triplets over a 3-D grid with resolution τ and randomly picking one configuration from each bin (see Appendix D for criteria to choose τ).

This procedure is repeated until the number of dictionary atoms reach the desired number of subarrays, $n = N_S$. All arrays in the dictionary $\mathcal{C}$ obtained from this algorithm satisfy the AOS$(\boldsymbol{C})$ constraint by construction. This simplifies the constrained multi-objective optimization in (5) to

$$\boldsymbol{C}^* = \arg\min_{\boldsymbol{C} \in \mathcal{C}} f(\boldsymbol{C}, \mathbb{1}_N) \tag{9}$$

B. Iterative Placement Refinement

In the second stage, we try to improve the cost function (9) by applying small local perturbations (within a bin of the grid G) to the subarray positions obtained from the combinatorial search in the first stage, as described in Algorithm 2.

Algorithm 2 Local Refinements

1: INITIALIZE: $C = C^{init}$; B = oversampled bin.
2: **while** $n < N_{ref}$ **do**
3: **for** $i = 1$ **to** N_s **do**
4: LIST Find positions available for adjustment, $V_i = \Im_B(\boldsymbol{C} \setminus C_i)$
5: CORRECT: Select position with least cost $C_i \leftarrow \min_{\boldsymbol{b} \in V_i} f\left(\{(\boldsymbol{C} \setminus C_i), \boldsymbol{b}\}, \mathbb{1}\right)$
6: **end for**
7: $n = n + 1$
8: **end while**
9: Return $\boldsymbol{C}$

Figure 6 shows a sample of how costs are minimized using sequential refinement over $N_s = 8$ subarrays. After running few iterations, the final array has 0.8 dB lower MSLL, while keeping other beam attributes relatively unchanged.

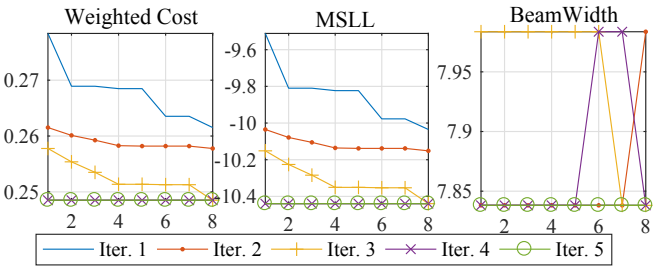


Fig. 6. Objective costs variation over iterative refinements.

C. Computational Complexity

The complexity of this approach is dominated by the construction of the prefix tree dictionary $\mathcal{C}$ in the first stage. Our algorithm progresses by growing leaf nodes of the tree in a breadth first fashion until a tree depth of N_s is reached. In each step, the key bottleneck lies in pruning the temporary dictionary $\hat{\mathcal{C}}^n$ which lists all possible vacancies for every leaf node of existing tree. The number of operations and space can be upper bounded as follows (see Appendix E for details):

$$T_{Algorithm1} = \sum_{n=1}^{N_s-1} |\hat{\mathcal{C}}^{n+1}| \leq \sum_{n=1}^{N_s-1} |G||\mathcal{C}_{max}^n| \sim O\left(|G|^4 N_s^4\right)$$
$$S_{Algorithm1} = \max_{n \in [1, N_s]} |\mathcal{C}^n| \leq |\mathcal{C}_{max}^n| \sim O\left(|G|^3 N_s^3\right)$$

The resulting dictionary is independent of cost function weights $[\alpha, \beta, \gamma]$, and represents a thinned version of the entire design space. Thus, optimized arrays for different weights can be evaluated by solving (9). The solution of (9) has $O(|\mathcal{C}|)$ time complexity, assuming that cost evaluation for each array configuration takes constant time. The second stage of the algorithm requires cost evaluation of fine perturbations over an oversampled bin of size $|B|$ around each subarray, repeated N_{ref} times. This stage requires $T_{Algorithm2} = |B| N_{ref} N_s$ operations using constant space, which is significantly lower compared to first stage. Hence, the overall algorithm complexity is $O(|G|^4 N_s^4)$ in time and $O(|G|^3 N_s^3)$ in space.

We note that the polynomial complexity of proposed algorithm is significantly better compared to exhaustive search, which exhibits exponential complexity $O(|G|^{N_s})$. For our running example, the proposed algorithm's complexity is $T_{Algorithm1} = (400 \times 8)^4 \approx 10^{14}$ compared to the complexity of exhaustive search, $T_{Exhaustive} = (400)^8 > 10^{20}$.

IV. Estimation-Theoretic Benchmarks

We now seek to evaluate the efficacy of our sparse designs for the canonical application of 2D DoA estimation. We compare different array designs in terms of estimation-theoretic bounds as well as simulated performance using a super-resolution algorithm. For clarity in exposition, from here onwards we overload $\boldsymbol{u} \triangleq [u, v]$ to denote the DoA.

A. Signal Model

We model the received signal from K sources in the scene with distinct DoAs $\Theta = [\boldsymbol{u_1}, \boldsymbol{u_2}, \cdots, \boldsymbol{u_k}]$ as

$$\boldsymbol{x} = \sum_{j=1}^{K} \alpha_j \boldsymbol{s}(\boldsymbol{u}_j) + \boldsymbol{z} \tag{10}$$

where $\boldsymbol{s(u_j)} = \left[e^{jk\boldsymbol{u_j^T d_1}}, \ldots, e^{jk\boldsymbol{u_j^T d_N}}\right]^T$ is the array response, $\boldsymbol{z} = [z_1, \ldots, z_N]^T$ is complex white noise such that $\mathbb{E}(zz^H) = \sigma^2 I_N$, and $\{\alpha\}_{j=1}^K$ are complex gains which are unknown deterministic constants. The joint probability density of received signal conditioned on $(\Theta, \{\alpha\}_{j=1}^K)$ is given by,

$$p(\boldsymbol{x}|\Theta, \boldsymbol{\alpha}) = \prod_{\boldsymbol{u}_j \in \Theta} \frac{1}{\pi^N \sigma^2} \exp\left(-\frac{\|\boldsymbol{x} - \alpha_j \boldsymbol{s}(\boldsymbol{u}_j)\|^2}{\sigma^2}\right) \tag{11}$$

For any DoA estimator $\hat{\Theta}$, the covariance of estimation error is defined as,

$$\boldsymbol{R}_\epsilon(\hat{\Theta}) = \mathbb{E}\left[\sum_{i=1}^{K} (\boldsymbol{u} - \hat{\boldsymbol{u}}_{\boldsymbol{i}})(\boldsymbol{u} - \hat{\boldsymbol{u}}_{\boldsymbol{i}})^T\right]$$

$\boldsymbol{R}_\epsilon$ can be geometrically interpreted by its trace $\sqrt{tr(R_\epsilon)}$ which represents the expected overall Root mean square error (RMSE) in DoA estimation (see Appendix A). We use this measure to compare the performance of array designs in Section V. For single source case ($K = 1$), the joint maximum likelihood estimator of $\boldsymbol{u}$ and α yields a noncoherent estimator for $\boldsymbol{u}$ as follows:

$$\hat{\boldsymbol{u}}_{ML} = arg\max_{\boldsymbol{u}} \left|\boldsymbol{s(u)^H x}\right|^2 \tag{12}$$

For this case, we derive the Cramer Rao (CRB) and Ziv-Zakai (ZZB) bounds on $\boldsymbol{R}_\epsilon$ to assess the best possible estimation accuracy of different designs. Although derived for single source case, we use these bounds for multiple source case as well to compare DoA estimation performance.

1) Cramér Rao Bound : The Bayesian Cramér Rao Bound for this signal model is given by [24]:

$$CRB(\boldsymbol{R_\epsilon}) = (\boldsymbol{J_F} + \boldsymbol{J_P})^{-1} \tag{13}$$

where $\boldsymbol{J_F}, \boldsymbol{J_P}$ denote the Fisher Information Matrix (FIM) contributions from the observation and the prior distribution of DoA respectively.

$$(\boldsymbol{J_F})_{ij} = -\mathbb{E}_{\boldsymbol{x},\boldsymbol{u}}\left[\frac{\partial^2 l(\boldsymbol{x}|\boldsymbol{u})}{\partial u_i \partial u_j}\right], (\boldsymbol{J_P})_{ij} = -\mathbb{E}_{\boldsymbol{u}}\left[\frac{\partial^2 l(\boldsymbol{u})}{\partial u_i \partial u_j}\right]$$

where $l(\boldsymbol{x}|\boldsymbol{u})$ and $l(\boldsymbol{u})$ are the conditional log likelihood and prior log likelihoods, respectively. In addition, the following regularity condition needs to be satisfied,

$$\mathbb{E}_{\boldsymbol{x},\boldsymbol{u}}\left[\frac{\partial l(\boldsymbol{x}|\boldsymbol{u})}{\partial \boldsymbol{u}}\right] = \boldsymbol{0}$$

$$\mathbb{E}_{\boldsymbol{x},\boldsymbol{u}}\left[jk\sum_{i=1}^{N} \boldsymbol{d}_i\left(x_i e^{jk\boldsymbol{u}^T\boldsymbol{d}_i} - x_i^* e^{-jk\boldsymbol{u}^T\boldsymbol{d}_i}\right)\right] = \boldsymbol{0}$$

$$jk\left(\alpha\sum_{i=1}^{N}\boldsymbol{d}_i - \alpha^*\sum_{i=1}^{N}\boldsymbol{d}_i\right) = jk(\alpha - \alpha^*)\sum_{i=1}^{N}\boldsymbol{d}_i = \boldsymbol{0}$$

In order to always satisfy this condition, we enforce the array element positions to be centered i.e., $\sum_{i=1}^{N} \boldsymbol{d}_i = \boldsymbol{0}$.

For a single source, the FIM is given by,

$$J_F = -\frac{1}{\sigma^2}\mathbb{E}\left[\frac{\partial \boldsymbol{s}(\boldsymbol{u})}{\partial \boldsymbol{u}}^H \frac{\partial \boldsymbol{s}(\boldsymbol{u})}{\partial \boldsymbol{u}}\right] \tag{14}$$

$$= 2k^2\gamma \boldsymbol{D}^T\boldsymbol{D} \tag{15}$$

which depends only on the element positions, $\boldsymbol{D}$ and Signal to Noise ratio (SNR) ($\gamma = |\alpha|^2/\sigma^2$). Assuming the DoA prior to be uniformly distributed in the ROI ($\theta \leq 30°$), the prior FIM simplifies to $\boldsymbol{J_P} = 1.343\boldsymbol{I}_2$.

2) Ziv-Zakai Bound: The CRB is a local bound, which accounts for estimation performance dependent on mainbeam, hence it is only useful at high SNR. In order to better characterize the estimation performance of Sparse arrays at low SNR, we calculate the Ziv-Zakai Bound (ZZB) which incorporates the effect of sidelobes and predicts the threshold behavior. For any directional vector $\boldsymbol{a} = [\cos\xi, \sin\xi]^T$, the ZZB is given by [25]

$$\boldsymbol{a^T R_\epsilon a} \geq \int_0^\infty \mathcal{V}\left\{\max_{\boldsymbol{\delta}:\boldsymbol{a^T\delta}=h} \int A(\boldsymbol{u},\boldsymbol{\delta}) P_e(\boldsymbol{u},\boldsymbol{\delta}) d\boldsymbol{u}\right\} hdh$$

where, $A(\boldsymbol{u},\boldsymbol{\delta}) = \min\{p(\boldsymbol{u}), p(\boldsymbol{u}+\boldsymbol{\delta})\}$, $\mathcal{V}(.)$ is the valley filling function and $P_e(\boldsymbol{u},\boldsymbol{\delta})$ is error probability of the following vector parameter binary detection problem,

$$H_0 : \hat{\boldsymbol{u}} = \boldsymbol{u}; \quad Pr(H_0) = \frac{1}{2}, \boldsymbol{x} \sim p(\boldsymbol{x}|\boldsymbol{u})$$

$$H_1 : \hat{\boldsymbol{u}} = \boldsymbol{u} + \boldsymbol{\delta}; \quad Pr(H_1) = \frac{1}{2}, \boldsymbol{x} \sim p(\boldsymbol{x}|\boldsymbol{u}+\boldsymbol{\delta})$$

This error probability can be lower bounded by the minimum probability of error of the following optimal non-coherent detector:

$$\text{Decide}(\boldsymbol{u}) = \begin{cases} \boldsymbol{u} & \text{if } \rho_1 > \rho_2 \\ \boldsymbol{u}+\boldsymbol{\delta} & \text{if } \rho_1 < \rho_2 \end{cases}$$

where $\rho_1 = |\boldsymbol{x^H s(u)}|$ and $\rho_2 = |\boldsymbol{x^H s(u+\delta)}|$. Given $\boldsymbol{u} = \boldsymbol{u_0}$, ρ_1, ρ_2 are rician distributed with scale parameter $s = \sigma^2/M$ and non-centrality parameter $\nu = |\alpha|N, |\alpha R(\boldsymbol{\delta})|N$ respectively where $R(\boldsymbol{\delta}) = R(\delta_x, \delta_y)$ is the beampattern from (3). The error probability is given by [26]

$$\begin{aligned} P_{nc}(\boldsymbol{u},\boldsymbol{\delta}) &= \frac{1}{2}\left(Pr\left(\rho_1 < \rho_2|\boldsymbol{u}\right) + Pr\left(\rho_1 > \rho_2|\boldsymbol{u}+\boldsymbol{\delta}\right)\right) \\ &= Pr\left(\rho_1 < \rho_2|\boldsymbol{u}\right) \\ &= Q_1(a,b) - \frac{1}{2}e^{-\frac{a^2+b^2}{2}} I_0(ab) \end{aligned} \tag{16}$$

where,

$$a = \sqrt{\frac{\gamma N}{2}\left(1 - \sqrt{1 - |R(\boldsymbol{\delta})|^2}\right)}$$

$$b = \sqrt{\frac{\gamma N}{2}\left(1 + \sqrt{1 - |R(\boldsymbol{\delta})|^2}\right)}$$

which is not a function of $\boldsymbol{u}$. For ROI in our case, the maximum error $h^{(\max)} = \left(\boldsymbol{a}^T\boldsymbol{\delta}\right)^{(\max)} = 1$. Note that for a uniformly distributed DoA in spherical coordinates (θ, ϕ), the distribution of $\boldsymbol{u}$ is not uniform. However for simplicity of analysis, we make the assumption that $\boldsymbol{u}$ is uniformly

distributed on a circular disc. Hence, the ZZB expression simplifies to

$$\boldsymbol{a^T R_\epsilon a} \geq \int_0^1 \mathcal{V}\left\{\max_{\boldsymbol{\delta}:\boldsymbol{a^T\delta}=h} \int A(\boldsymbol{u})d\boldsymbol{u}P_{nc}(\boldsymbol{\delta})\right\} hdh$$

$$ZZB(\boldsymbol{a^T R_\epsilon a}) = \int_0^1 \mathcal{V}\left\{\max_{\boldsymbol{\delta}:\boldsymbol{a^T\delta}=h} P_{nc}(\boldsymbol{\delta})\right\} hdh \quad (17)$$

The maximum error probability over all directions $\boldsymbol{\delta}$ cannot be expressed as closed form expression. However, due to the monotonicity of Marcum's Q function, $Q_1(.)$ and Bessel function of 0^{th} order, $I_0(.)$, the error probability in (16) is maximized only when $R(\boldsymbol{\delta})$ is maximized. Therefore, for each values of h, we compute the $\max_{\boldsymbol{\delta}:\boldsymbol{a^T\delta}=h} |R(\boldsymbol{\delta})|$ numerically by searching over a discrete set of points on the line segment $\boldsymbol{a^T\delta} = h$ and substitue in (17).

B. DoA estimation algorithm

Grid-based sparse estimation for a set of DoAs models the the received signal (10) as follows:

$$\boldsymbol{x} = \boldsymbol{S(\Psi)b} + \boldsymbol{z} \quad (18)$$

where $\boldsymbol{S(\Psi)} = \left[\boldsymbol{s(u_1)}\cdots\boldsymbol{s(u_{|\Psi|})}\right]$ contains the array response at discretized set of DoAs $u_i \in \boldsymbol{\Psi}$ as columns. The nonzero entries in $\boldsymbol{b}$ point to presence of target in the corresponding DoA in Ψ. The DoA and gain pair $(\hat{\boldsymbol{u}}_i, \hat{\alpha}_i)_{i=1}^K$ can be estimated by jointly minimizing the residual power,

$$T(\hat{\boldsymbol{u}}, \hat{\alpha}) = \left\| \boldsymbol{x} - \sum_{j=1}^{K} \hat{\alpha}_j \boldsymbol{s}(\hat{\boldsymbol{u}}_j) \right\|^2$$

The NOMP algorithm summarized below provides a two stage estimator:

1) *Detection*: Using precomputed $\boldsymbol{S(\Psi)}$, coarse estimates of DoA and complex gain are obtained

$$\hat{\boldsymbol{u}} = arg\max_{\boldsymbol{u}\in\boldsymbol{\Psi}} |\boldsymbol{s}(\boldsymbol{u})^H \boldsymbol{x}|^2$$
$$\hat{\alpha} = \boldsymbol{s}(\boldsymbol{u})^H \boldsymbol{x}/N$$

2) *Refinement*: The estimates are refined using the Newton method:

$$\hat{\boldsymbol{u}}' = \hat{\boldsymbol{u}} - \left(H_\nabla T(\hat{\boldsymbol{u}}, \hat{\alpha})\right)^{-1} \nabla T(\hat{\boldsymbol{u}}, \hat{\alpha}) \quad (19)$$
$$\hat{\alpha}' = \boldsymbol{s}(\hat{\boldsymbol{u}}')^H \boldsymbol{x}/N \quad (20)$$

where $H_\nabla T$ and ∇T denote the Hessian and gradient of $T(\boldsymbol{u}, \alpha)$ with respect to $\boldsymbol{u}$ at current estimate $(\hat{\boldsymbol{u}}, \hat{\alpha})$ (see [27] for details).

The algorithm is repeated with the residual signal, $\boldsymbol{r} = \boldsymbol{x} - \hat{\alpha}'\boldsymbol{s}(\hat{\boldsymbol{u}}')$ to estimate other DoAs. The *refinement* steps are repeated after each new detection for all DoAs in a cyclic manner for few rounds to improve accuracy.

The algorithm yields K_{est} DoA estimates, with estimation performance degrading when K_{est} does not match the true number of DoAs, K_{DoA}. Hence, in order to evaluate the arrays independent of such errors, we implement both algorithm where $K = K_{DoA}$ is known.

We also run extensive simulations with another state of the art algorithm, BPDN [17], with default parameters and 5 refinement stages. The computational complexity of BPDN is significantly higher than that of NOMP. The grid $\boldsymbol{\Psi}$ needs to be adapted at each iteration for BPDN, depending on the DoAs, whereas it remains fixed for NOMP $\boldsymbol{S(\Psi)}$, and can be precomputed, which makes it suitable for faster implementation in large arrays. Since the NOMP algorithm also yields somewhat better estimation accuracy than BPDN, we only present results obtained with NOMP here.

V. Numerical Results

A. Design of arrays

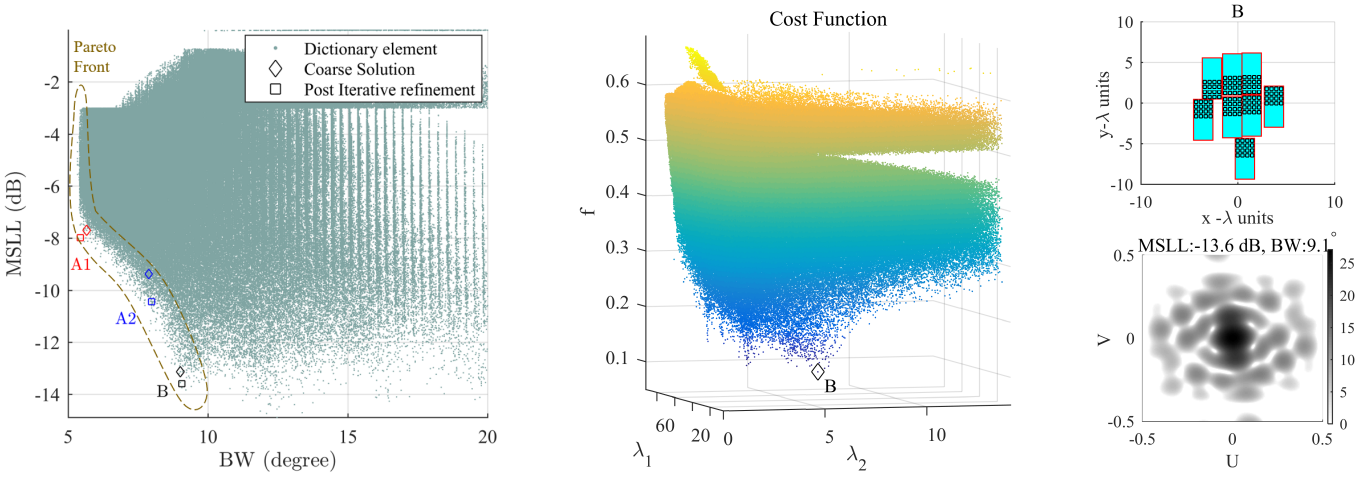
Using the combinatorial search algorithm, we create a search space of array configurations, $\boldsymbol{C} \in \mathcal{C}$ of size $|\mathcal{C}| = 657,000$ for $N_{sub} = 8$ subarrays. Figure 7a shows the values of two major objectives, `MSLL`, `BW` over this space. The beamwidth is improved by spreading out the subarrays, but this typically leads to a deterioriation in the MSLL. Figure 7a clearly shows the Pareto front corresponding to the multiple solutions of (9), corresponding to different relative weights, trading off these opposing objectives. Figure 7b shows the weighted cost function obtained for an example set of weights $(\alpha = 0.1, \beta = 0.5, \gamma = 0.1)$ as a function of the eigenvalues of the array covariance matrix. We observe that for a similar set of eigenvalues, many of the solutions we explore (shown in yellow) are substantially worse in terms of the weighted cost than the solutions shown in blue from which we choose our solutions, indicating the complexity of the optimization landscape. Choosing one more variable, the variance of the element locations $\psi(\boldsymbol{C})$, for binning is therefore crucial for exploring this landscape more thoroughly. Figure 7c shows the array B obtained using the proposed algorithm, and its beampattern. We mark the locations of B, and of two other Pareto optimal designs, $A1$ and $A2$ (to be discussed shortly) along the Pareto front in Figure 7a.

As we change the relative values of weights α, β and γ in exploring the Pareto front, we can make the following observations regarding the corresponding subarray placements:

- For a larger relative value of α (more importance given to beamwidth), the subarrays are widely distributed over the available aperture area.
- When we increase the relative value of β to suppress MSLL, the array becomes restricted to a smaller area.
- The relative weight of γ, which corresponds to the objective of reducing eccentricity, affects both the shape of the main lobe and the positions of the sidelobes. For positive γ, the solution is not expected to lie on the `BW-MSLL` Pareto front boundary. Indeed, Figure 7a shows that the solutions from first stage dictionary search lie slightly away from this boundary. However, the second stage iterative refinement reduces this gap.

Based on these observations, we set the weights to obtain following sample array configurations:

1) *A1*: Primary emphasis is given towards minimizing beamwidth by setting $\alpha = 1$, $\beta, \gamma = 0.1$.
2) *A2*: In this case, we emphasize all beam attributes by setting all weights equal to 1.



(a) Beam attributes of array dictionary $\mathcal{C}$, Sample arrays on Pareto optimal front obtained from first stage dictionary search ($\diamond$) and second stage iterative placement refinement ($\square$).

(b) Weighted cost function $f(\boldsymbol{C}, \mathbb{1}_N), \boldsymbol{C} \in \mathcal{C}$ for weights $(0.1, 0.5, 0.1)$ as a function of the eigenvalues of the array covariance matrix $\Sigma(C)$.

(c) Array configuration & beam-pattern for solution B.

Fig. 7. Pareto-front exploration and multi-objective optimization.

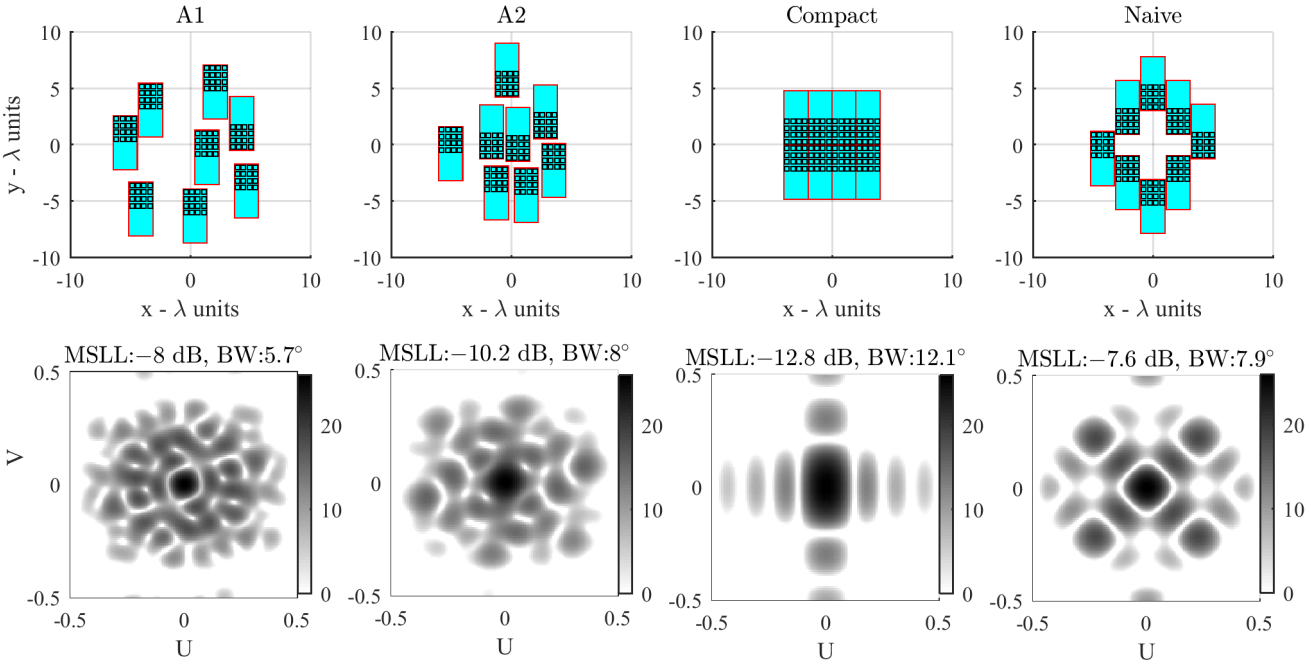


Fig. 8. Beam patterns (Bottom row) for designed (Left half) & benchmarking (right half) arrays.

TABLE I
SAMPLE ARRAY CONFIGURATIONS

Shape	α	β	γ	MSLL	BW	ecc
A1	1	0.1	0.1	-8 dB	5.7°	0
A2	1	1	1	-10.2 dB	8°	0
Compact	-	-	-	-12.8 dB	12.1°	0.7
Naive	-	-	-	-7.6 dB	7.9°	0.0

We compare these array designs against two simple array configurations, 1) a "compact" array where subarrays are placed together such that overall element pattern becomes a uniform rectangular array, 2) a "naive" array where subarrays are spread along a diamond shape such that its resultant beamwidth is equal to that of *A2*. Table I lists the weights and resulting beam attributes of these arrays.

Figure 8 shows the array designs obtained using our optimization approach and their beam patterns. The *A2* array has a sharp beamwidth and only 2.6 dB worse MSLL compared to the compact array. On the other hand, a naive sparse array with circular arrangement of subarrays yields 2.6 dB higher MSLL compared to *A2* for similar beamwidth. Our designs *A1*, *A2* exhibit several small sidelobes (the highest sidelobe for *A2* is -10.2 dB), whereas the naive array exhibits fewer but

more pronounced sidelobes. Since large sidelobes and grating lobes can cause large errors in DoA estimation, we expect our designs to yield better estimation performance, which is borne out by the results presented in the next section.

We note that the sidelobe levels and locations are primarily dependent on the super-array (i.e., the location of subarray centers $\boldsymbol{C}$), while the subarray pattern mainly affects the large-scale beampattern. Therefore, we obtain similar solutions, with similar beam characteristics, using our optimization approach with small variations in the subarray element patterns (e.g, replacement of a rectangular pattern with a plus pattern). Such results are not reported here due to space constraints.

B. Comparison of Estimation Performance

We evaluate the arrays based on their DoA estimation accuracy at different SNRs for both single and multiple source cases. We use the RMSE in estimating DoA for comparison which is given by $\bar{\epsilon} = \sqrt{\mathbb{E}[|\hat{\boldsymbol{u}} - \boldsymbol{u_0})|^2]} = \sqrt{tr(\boldsymbol{R_\epsilon})/2}$.

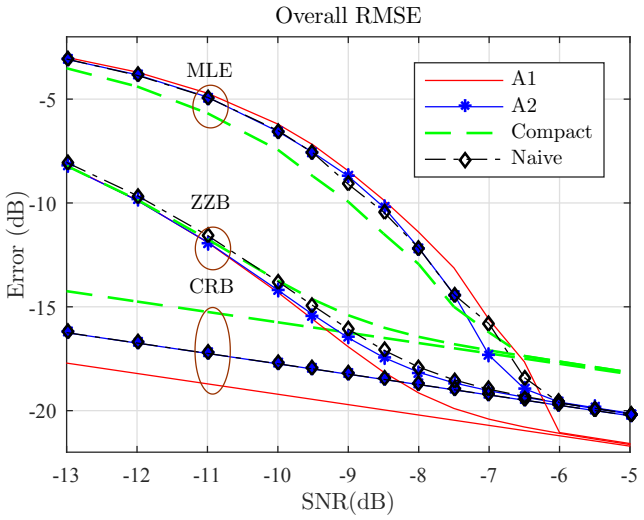


Fig. 9. Comparison of estimation theoretic bounds for arrays.

1) Estimation bounds: The Cramér Rao Bound is evaluated using (13), $CRB(\bar{\epsilon}) = \sqrt{tr(CRB(\boldsymbol{R_\epsilon}))/2}$. The Ziv Zakai Bound is evaluated using (17),

$$ZZB(\bar{\epsilon}) = \sqrt{ZZB(\boldsymbol{a_1^T R_\epsilon a_1}) + ZZB(\boldsymbol{a_2^T R_\epsilon a_2})}$$

where $\boldsymbol{a_1}, \boldsymbol{a_2}$ denote the directions of maximum and minimum beamwidths of the array. We also computed the ML estimation (MLE) error by Monte Carlo simulation using (12) with an overcomplete dictionary of array responses. Figure 9 shows the CRB, ZZB and MLE curves for all the arrays. CRB is proportional to beamwidth (see Appendix B for details). The ZZB bound converges to CRB at the so-called "ZZB threshold" SNR: when the SNR is below this threshold, far-ambiguities in DoA estimation caused by large sidelobes dominate the MSE. The tradeoff between beamwidth and MSLL is thus expected to translate to one between CRB (better with smaller beamwidth) and ZZB threshold (worse with larger MSLL). Thus, as expected, "*A1*" array achieves the lowest CRB, followed by "*naive*" and "*A2*" with equal CRB, while "*Compact*" array has the largest beamwidth and hence highest CRB. The trend in MSLL is weakly reflected in the ZZB thresholds (for a single target, sidelobes do limited damage): the degradation in ZZB threshold, relative to that of the compact array for the optimized arrays (*A1, A2*) is less than 1 dB, while the gain in CRB due to smaller beamwidth is 4 dB and 2 dB, respectively. The MLE error curve also agrees with the threshold behavior predicted by ZZB. We see in the next set of results, however, that the size of the sidelobes becomes much more important when we consider multiple targets.

2) Estimation algorithm performance: We obtain DoA estimates using the *NOMP* algorithm [18], [27] with a known number of sources to compare the best case performance of these arrays. (The NOMP algorithm also performs as well as the brute force MLE for a single target discussed earlier– results omitted here.) The RMSE is evaluated across $N = 1024$ DoAs uniformly sampled over the ROI (spherical cap of half angle $30°$). For evaluating the estimation performance in presence of multiple targets, ($K = 5$) we compute the RMS error in the DoA estimate for a primary target fixed at broadside, while interfering targets are distributed uniformly in ROI at separation of $\Delta\boldsymbol{u} \geq 0.16$ or $\Delta\theta \geq 9.2°$ away from primary target. This separation is imposed because the estimation problem is ill-posed for DoAs in close proximity. For uniform arrays, the minimum separation is typically defined with respect to the DFT bin size (e.g. $\Delta_{\text{DFT}} = 2\pi/L$ for an L-element linear array). Since this quantity cannot be defined for non-uniform planar arrays, we choose a minimum separation halfway between the RMS beamwidths of the "compact" and sparse arrays, to capture the effect of both local errors and far ambiguity errors due to sidelobes.

In addition to RMSE vs SNR curves, we also analyze the distribution of error magnitudes. The complementary cumulative distribution (CCDF) of the estimation errors is used to compare the "outage probability" corresponding to too large an error, which captures the impact of large sidelobes.

- With multiple sources, the estimation accuracy is degraded by interference from other sources, and RMSE does not converge to the single-target CRB. Figure 10 shows the estimation performance for strong and weak interference.
  1) Weak interference: When the interfering sources are 6 dB weaker than primary target, sparse arrays offer more than 5 dB SNR gain compared to compact arrays for $SNR > -5$ dB. Also, the difference between *A2, naive* array widens to about 1 dB in the threshold region indicating the benefit of suppressing sidelobes.
  2) Strong interference: When interfering sources have same magnitude, RMSE severely degrades for both arrays with high sidelobes (*A1, Naive*) as well as arrays with high beamwidth (*Compact*). On the other hand, *A2* has lowest RMSE at $SNR > -5$ dB because of the dual benefit of small beamwidth and lower sidelobes.
- The increase in estimation errors at high SNR is attributed to ambiguity errors from sidelobes, hence the overall

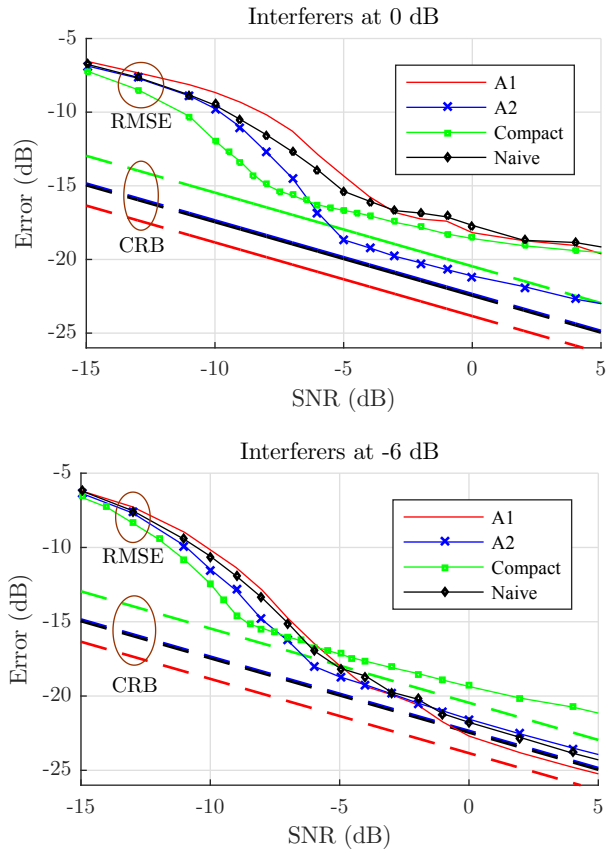


Fig. 10. Estimation accuracy with multiple targets.

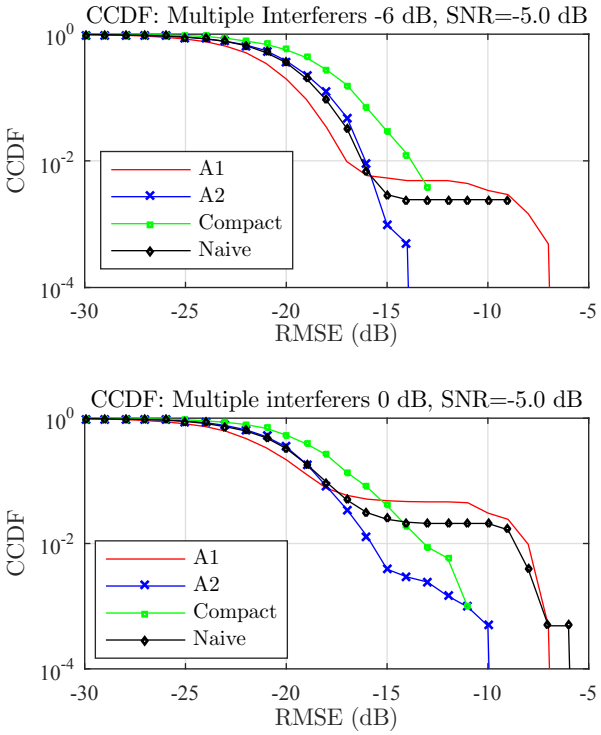


Fig. 11. CCDF of estimation errors in multiple targets.

sidelobe suppression for the arrays can be compared using the distribution of these error magnitudes. Figure 11 shows the CCDF curves of all arrays at SNR=-5 dB. The initial curvature of these curves (RMSE upto -22 dB) is expected to depend on local errors, hence the rate of change follows same order as CRB which is *A1* $>$ *A2* = *naive* $>$ *compact*. But the curvature reverses order at higher RMSE indicating the tradeoff with far-errors. We can see that *A2* achieves lower outage probability in both scenarios (e.g. for RMSE threshold set to -15 dB) as it strikes a balance between near and far errors. In contrast, both *A1* and *naive* exhibit high outage probability due to frequent far ambiguity errors caused by higher sidelobes.

Therefore, depending on the expected magnitude of interferers either one of the designed arrays with suitable sidelobe suppression can be selected. For a desired beamwidth reduction our design algorithm yields an array superior to a naively designed sparse array.

VI. Compressive Estimation

We now evaluate the arrays for sparse estimation using compressive measurements at each subarray given by:

$$\boldsymbol{y} = \boldsymbol{\Phi x}$$

where $\boldsymbol{x}$ is the full measurement from (18) and $\boldsymbol{\Phi} = \text{diag}(\boldsymbol{\Phi}_1, \cdots, \boldsymbol{\Phi}_{N_s})$ is the $M \times N$ measurement matrix consisting of the subarrays measurement matrices as its block diagonals. Each subarray takes M_i compressive measurements with an independent $\boldsymbol{\Phi}_i \in \mathbb{C}^{M_i \times N_e}$ whoose elements are chosen uniformly and independently from QPSK samples $\frac{1}{\sqrt{M_i}}\{\pm 1, \pm j\}$. In addition, columns of $\boldsymbol{\Phi}_i$ have unit norm to preserve signal norm on average ($\mathbb{E}\left[||\Phi \boldsymbol{S}(\boldsymbol{u})||^2\right] = ||\boldsymbol{S}(\boldsymbol{u})||^2$) while scaling noise variance by N/M. The underlying DoA, $\boldsymbol{u}$ can be extracted by minimizing the ML cost:

$$\left\| \boldsymbol{y} - \Phi \sum_{j=1}^{K} \hat{\alpha_j} \boldsymbol{s}(\hat{\boldsymbol{u}}_j) \right\|^2$$

The efficacy of compressive parameter estimation in AWGN depends on preserving the geometric structure of the parameterized signals [20]. Specifically, if Φ satisfies the $2K$ isometry property for discretized basis $\boldsymbol{S}(\boldsymbol{\Psi})$ [20],

$$C(1-\epsilon) \leq \frac{|\Phi \boldsymbol{S}(\boldsymbol{\Psi})\boldsymbol{b}|^2}{|\boldsymbol{S}(\boldsymbol{\Psi})\boldsymbol{b}|^2} \leq C(1+\epsilon) \tag{21}$$

where C is a constant for any arbitrarily chosen $2K$ sparse vector $\boldsymbol{b}$, the performance of the compressive system follows that for the original system, except for an SNR penalty of M/N. Figure 12 shows the minimum and maximum values of this ratio over 10^6 random realization of 8 sparse $\boldsymbol{b}$ for sparse array. The ratio is within $[-5, 3]$ dB for $M > 32$ signifying that 32 compressive measurements are sufficient to estimate $K = 4$ DoAs.

Figure 13 shows estimation performance with $M = 32$ compressive measurements collected across eight subarrays ($M_i = 4, i \in \{1..8\}$). Comparing with Figure 9, we observe that the estimation algorithms preserve the same characteristics

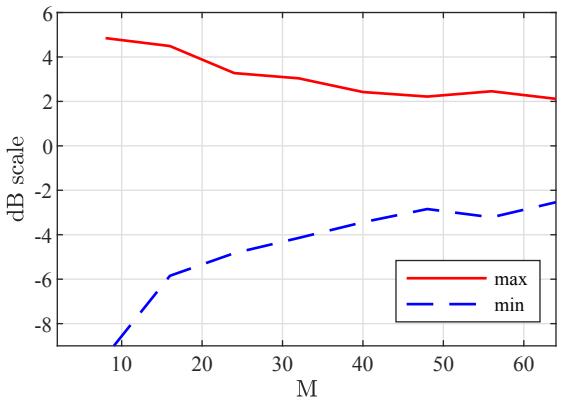


Fig. 12. Maximum and minimum values of ratio in (21) for sparse array.

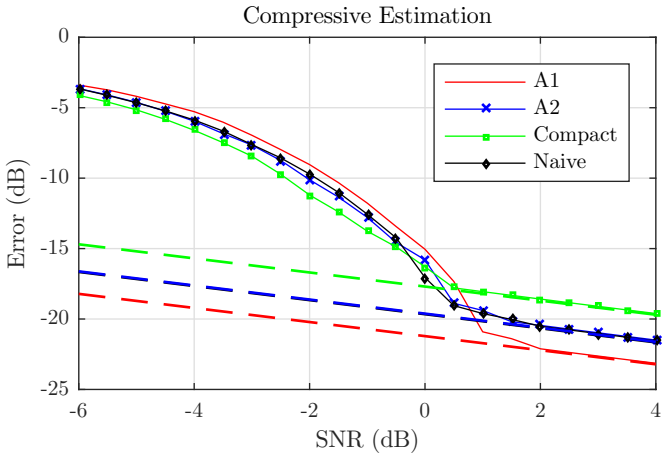


Fig. 13. Estimation performance with Compressive measurements.

as with full measurements with approximately 6 dB SNR penalty as expected ($N/M = 4$).

VII. Conclusions

Our results demonstrate that trading off beam width versus side lobes when synthesizing a large effective aperture does indeed produce performance gains in bearing estimation. Compared to a compact placement of subarrays, an optimized sparse placement produces smaller mean squared error because of its smaller beam width. Compared to a naive sparse placement, the control of side lobes via our optimized placement produces the most significant gains when estimating the bearing for multiple sources.

Exploring the application of our framework for communications, where transmit and receive beamforming gains are fixed by the number of elements, but control of beam width and sidelobes affects interference, is an interesting direction. In the context of sensing, our work may be viewed as design of an individual sensor which can be placed within a more comprehensive architecture, such as a network of sensors for localization and tracking.

In our tiled architecture, the locations of antenna elements in a subarray are fixed once we specify the location of the subarray center. This constraint makes it difficult to adapt the extensive literature on sparse array optimization, which typically considers elements that can be freely placed, for our present purpose. The difficulty is compounded by the lack of closed form expressions for beam attributes of interest. However, since the objectives are only mildly dependent on the configuration of elements *within* a subarray, it might be possible to simplify the optimization problem, and possibly adapt ideas from the literature on sparse array optimization. This is an interesting direction for future research, especially given the importance of tiled architectures in realizing a large aperture leveraging low-cost hardware for subarrays with a moderate number of elements.

Appendix A
Mean Square Error in 2D DoA Estimation

For 2D DoA estimation, the error along any given angle ξ is given by $\boldsymbol{a}^T\boldsymbol{R}_\epsilon\boldsymbol{a}$, where $\boldsymbol{a} = [\cos\xi, \sin\xi]^T$ is the directional cosine and $\boldsymbol{R}_\epsilon$ is the error covariance matrix. Assuming that $\{\nu_i, \boldsymbol{q_i}, i = 1, 2\}$, denote the eigenvalues and eigenvectors of $\boldsymbol{R}_\epsilon$, the MSE averaged over $\boldsymbol{a}$ (assume ξ uniform over $[0, 2\pi]$) is given by

$$\begin{aligned}\text{MSE} = \mathbb{E}_{\boldsymbol{a}}\left[\boldsymbol{a}^T\boldsymbol{R}_\epsilon\boldsymbol{a}\right] &= \nu_1\mathbb{E}_{\boldsymbol{a}}\left[\left|\boldsymbol{a}^T\boldsymbol{q_1}\right|^2\right] + \nu_2\mathbb{E}_{\boldsymbol{a}}\left[\left|\boldsymbol{a}^T\boldsymbol{q_2}\right|^2\right] \\ &= \nu_1\frac{||\boldsymbol{q_1}||^2}{2} + \nu_2\frac{||\boldsymbol{q_2}||^2}{2} = (\nu_1+\nu_2)/2 = \frac{1}{2}tr(\boldsymbol{R}_\epsilon)\end{aligned}$$

where we have used $\mathbb{E}[\cos^2\xi] = \mathbb{E}[\sin^2\xi] = \frac{1}{2}$.

Appendix B
2D Beamwidth & CRB

We define 2D beamwidth using the Taylor series expansion of beampattern $R_{\boldsymbol{u_o}}(\boldsymbol{u})$ around mainlobe $R_{\boldsymbol{u_o}}(\boldsymbol{0})$. Since the beampattern around the main lobe. and hence the beamwidth, is invariant to beamforming direction (see II-C), we assume $\boldsymbol{u_o} = \boldsymbol{0}$ without loss of generality, and drop the subscript: $R_{\boldsymbol{0}}(\boldsymbol{u}) \triangleq R(\boldsymbol{u})$. By taking the derivatives of (3), theTaylor series expanision up to second order is obtained as

$$R(\boldsymbol{u}) \approx R(\boldsymbol{0}) - \frac{k^2}{N}\boldsymbol{u}^T\boldsymbol{D}^T\boldsymbol{D}\boldsymbol{u} \tag{22}$$

We define Half Power Beam Contour (*HPBC*) as the closed contour around mainbeam with $\{\boldsymbol{u} : R(\boldsymbol{u}) = 0.5R(\boldsymbol{0})\}$, which is approximated as an ellipse using (22) as follows:

$$\boldsymbol{u}^T\boldsymbol{D}^T\boldsymbol{D}\boldsymbol{u} = \frac{N}{2k^2}R(\boldsymbol{0}) \tag{23}$$

Consider the eigendecomposition of $\boldsymbol{D}^T\boldsymbol{D}$ given by,

$$\boldsymbol{D}^T\boldsymbol{D} = \lambda_1\boldsymbol{p}_1\boldsymbol{p}_1^T + \lambda_2\boldsymbol{p}_2\boldsymbol{p}_2^T \qquad (\lambda_2 \geq \lambda_1)$$

The eigenvectors $\boldsymbol{p}_2, \boldsymbol{p}_1$ correspond to major and minor axis of *HPBC* ellipse respectively, and depend only on the element positions.

Figure 14 shows the mainlobe of a beam and the dotted shaded region represents its *HPBC* ellipse. $\breve{\boldsymbol{u}}_o, \breve{\boldsymbol{u}}^1, \breve{\boldsymbol{u}}^2$ correspond to unit vectors in the direction of main beam, vertex and co-vertex of the *HPBC* where, $\breve{\boldsymbol{u}} = [u, v, \sqrt{1-u^2-v^2}]$

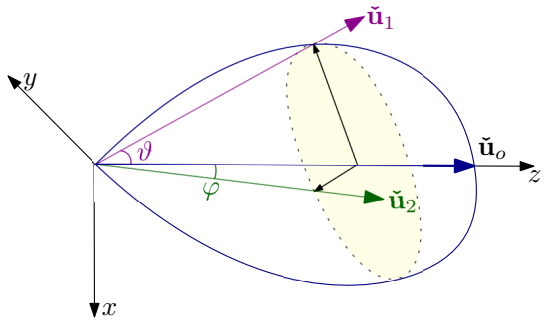


Fig. 14. 2D Beamwidth

is the unit vector towards directional cosine $\boldsymbol{u} = [u, v]$. These can be expressed as,

$$\check{\boldsymbol{u}}_o = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, \check{\boldsymbol{u}}_1 = \begin{bmatrix} \sin\vartheta\cos\phi_{\max} \\ \sin\vartheta\sin\phi_{\max} \\ \cos\vartheta \end{bmatrix}, \check{\boldsymbol{u}}_2 = \begin{bmatrix} \sin\varphi\cos\phi_{\min} \\ \sin\varphi\sin\phi_{\min} \\ \cos\varphi \end{bmatrix} \tag{24}$$

where $\phi_{\max}, \phi_{\min}$ are perpendicular azimuthal angles and ϑ, φ are the maximum and minimum beamwidth angles subtended from the major and minor axis of this ellipse to the mainbeam.

$$\mathrm{BW}_{\max} = \vartheta = \left(\frac{360}{\pi}\right)\cos^{-1}\left(\check{\boldsymbol{u}}_{\boldsymbol{o}}\check{\boldsymbol{u}}_{\boldsymbol{1}}\right) \tag{25}$$

$$\mathrm{BW}_{\min} = \varphi = \left(\frac{360}{\pi}\right)\cos^{-1}\left(\check{\boldsymbol{u}}_{\boldsymbol{o}}.\check{\boldsymbol{u}}_{\boldsymbol{2}}\right) \tag{26}$$

Substituting the major and minor axis from (24) in (23), we obtain

$$\lambda_1 \sin^2\vartheta = \frac{N}{2k^2}R(\mathbf{0}) \implies \sin\vartheta \approx \mathrm{BW}_{\max} \propto 1/\sqrt{\lambda_1}$$
$$\lambda_2 \sin^2\varphi = \frac{N}{2k^2}R(\mathbf{0}) \implies \sin\varphi \approx \mathrm{BW}_{\min} \propto 1/\sqrt{\lambda_2}$$

That is, the beamwidths along extremal directions are inversely proportional to the square roots of the eigenvalues of $\boldsymbol{D}^T\boldsymbol{D}$.

Relation to CRB: Using (14), the error covariance matrix is lower bounded by

$$\begin{aligned} \boldsymbol{R}_{\boldsymbol{\epsilon}} \geq \mathrm{CRB} = J_F^{-1} &= \frac{N}{2k^2\gamma}\left(\boldsymbol{D}^T\boldsymbol{D}\right)^{-1} \\ &= \frac{N}{2k^2\gamma}\left(\frac{1}{\lambda_1}\boldsymbol{p}_1\boldsymbol{p}_1^T + \frac{1}{\lambda_2}\boldsymbol{p}_2\boldsymbol{p}_2^T\right) \\ &\qquad \text{(using } (\lambda_2 \geq \lambda_1)\text{)} \end{aligned}$$

Using Appendix A, the MSE can be lowerbounded by

$$\begin{aligned} \mathrm{MSE} \geq \overline{\mathrm{CRB}} = \frac{1}{2}tr(J_F^{-1}) &= \frac{N}{4k^2\gamma}\left(\frac{1}{\lambda_1} + \frac{1}{\lambda_2}\right) \\ &= \frac{N}{4k^2\gamma}\left(\sin^2(\vartheta) + \sin^2(\varphi)\right) \\ &\propto \frac{(\mathrm{BW}^{\mathrm{DoA}})^2}{\mathrm{SNR}} \end{aligned}$$

where $\mathrm{BW}^{\mathrm{DoA}} = \sqrt{\mathrm{BW}_{\max}^2 + \mathrm{BW}_{\min}^2} = \sqrt{\vartheta^2 + \varphi^2}$ is defined as MSE beamwidth ($\sin\theta \approx \theta$ for small angles θ).

APPENDIX C
VACANCY SEARCH OPERATOR $\mathcal{T}$

Our reference subarray module shown in Figure 15 occupies space in addition to antenna elements. In order to keep element polarizations aligned, these modules can be placed in either *up* (0°) or *down* (180°) pose. We outline a procedure to list the vacant gridpoints $V_i = \mathcal{T}(\boldsymbol{C}_i^n)$ where the *new* subarray can be placed without overlapping with already placed *dormant* subarrays at $\boldsymbol{C}_i^n$. We define the subarray state as the center $\boldsymbol{c}$ of the element pattern and its *pose* ν, since vacancies depend on both parameters.

$$\tilde{\boldsymbol{c}} = \{\boldsymbol{c}, \nu\} \forall \boldsymbol{c} \in \boldsymbol{C}_i^n, V_i$$

The pose variable $\nu \in \{\nu^u, \nu^d, \nu^f\}$ denotes whether subarray can be placed in up only(ν^u), down only(ν^d) or free pose (ν^f, either up or down) at the location c. For a given set of *dormant* subarray states, $\tilde{\boldsymbol{C}}_i^n$ we identify all vacant states $\tilde{V}_i$ for placing the *new* subarray. Once a *new* subarray is placed, the states of all dormant subarrays are updated (e.g., a free pose may switch to an up pose if the down pose becomes infeasible).

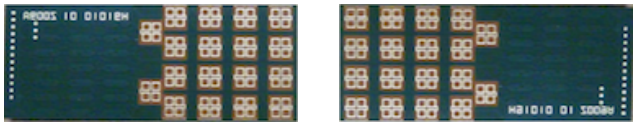


Fig. 15. The Subarray module and its two possible poses. Golden section are copper patch antennas on the green colored chip.

APPENDIX D
PERTURBATION OF ARRAY

In order to design a bin size for pruning array configurations, we analyze the effect of perturbing the location of a single array element on the eigenvalues of the array covariance matrix. Consider a small perturbation $\boldsymbol{\upsilon} = [\upsilon_x, \upsilon_y]$ added to i^{th} array element position: $\bar{\boldsymbol{d}}_i = \boldsymbol{d}_i + \boldsymbol{\upsilon}$. The covariance for the perturbed array is

$$\boldsymbol{\Sigma}_{\bar{\boldsymbol{D}}} = \left(\boldsymbol{D}^T\boldsymbol{D} + \boldsymbol{\upsilon}^T\boldsymbol{\upsilon} + 2\boldsymbol{d}_i^T\boldsymbol{\upsilon}\right)/N = \boldsymbol{\Sigma}_{\boldsymbol{D}} + \boldsymbol{\mathcal{G}} + 2\boldsymbol{\mathcal{H}}$$

where

$$\boldsymbol{\mathcal{G}} = \frac{1}{N}\begin{bmatrix} \upsilon_x^2 & \upsilon_x\upsilon_y \\ \upsilon_x\upsilon_y & \upsilon_y^2 \end{bmatrix}, \quad \boldsymbol{\mathcal{H}} = \frac{1}{N}\begin{bmatrix} \upsilon_x d_{xi} & \upsilon_x d_{yi} \\ \upsilon_y d_{xi} & \upsilon_y d_{yi} \end{bmatrix}$$

Using Weyl's inequality [28] for real symmetric matrices, the eigenvalue perturbation is bounded as

$$\begin{aligned} |\bar{\lambda}_i - \lambda_i| &\leq \|\boldsymbol{\mathcal{G}} + 2\boldsymbol{\mathcal{H}}\|_2 \leq \|\boldsymbol{\mathcal{G}}\|_2 + 2\|\boldsymbol{\mathcal{H}}\|_2 \\ &\leq \|\boldsymbol{\mathcal{G}}\|_F + 2\|\boldsymbol{\mathcal{H}}\|_F \end{aligned}$$

The frobenius norms of $\boldsymbol{\mathcal{G}}, \boldsymbol{\mathcal{H}}$ are

$$\|\boldsymbol{\mathcal{G}}\|_F = \left(\upsilon_x^2 + \upsilon_y^2\right)/N$$
$$\|\boldsymbol{\mathcal{H}}\|_F = R_i\sqrt{\left(\upsilon_x^2 + \upsilon_y^2\right)}/N$$

where $R_i = \sqrt{d_{xi}^2 + d_{yi}^2}$ is the distance of the i^{th} element from the array center. Hence, the overall variation of eigenvalues with variation $\Delta_e = \sqrt{\left(\upsilon_{xi}^2 + \upsilon_{yi}^2\right)}$ of the i^{th} element is

$$|\bar{\lambda}_i - \lambda_i| \leq \frac{(2R_i + \Delta_e)}{N}\Delta_e \tag{27}$$

Thus, the eigenvalues are more sensitive to perturbations in the locations of elements further from the center. For perturbations within one grid size used in our placement search algorithm, the eigenvalue of the subarray center covariance can vary at most by $\frac{(2R_i+\sqrt{2}\Delta_g)}{N_s}\sqrt{2}\Delta_g$, and we use this as a guideline for discretizing the eigenvalues for removing geometrically similar configurations.

APPENDIX E
ALGORITHM COMPLEXITY

The number of operations at the n^{th} iteration of Algorithm 1 is given by

$$T_n = |\hat{\mathcal{C}}^{n+1}| = \sum_{n=1}^{N_s-1}\sum_{i=1}^{|\mathcal{C}^n|}|V_i| \leq |G||\mathcal{C}^n| \tag{28}$$

where $\mathcal{C}^n$ denotes the set of leaf nodes at the n^{th} level of prefix tree, and $|V_i|$ is the number of vacancies for the i^{th} leaf node. The vacancies are a subset of a grid with cardinality $|G| = (\frac{2R_{max}}{\Delta_g})^2$ where R_{max} is the radius of the aperture and Δ_g is the grid resolution.

The maximum number of array configurations (which correspond to leaves of the prefix tree) in any given iteration is bounded by the maximum number of unique triplets $(\lambda_1, \lambda_2, \psi)$, which can be expressed as follows:

$$|\mathcal{C}^n| \leq |\mathcal{C}^n_{max}| = \frac{\lambda_1^{max}}{\tau_{min}}\frac{\lambda_2^{max}}{\tau_{min}}\frac{\psi^{max}}{\tau_{min}} \tag{29}$$

where $\lambda_1^{max}, \lambda_2^{max}, \psi^{max}$ represent the maximum value of each parameter and $\tau_{min} = \frac{2\Delta_g^2}{N_s}$ is the minimum bin resolution (using (27)). The eigenvalues of the array covariance are bounded by the maximum aperture radius $\lambda_1^{max} < R_{max}^2, \lambda_2^{max} < R_{max}^2$ and the array separation variance can be bounded as

$$\psi = \mathbb{E}\left[(l_{ij} - \mathbb{E}[l_{ij}])^2\right] \leq R_{max}^2 \qquad (\because 0 \leq l_{ij} \leq 2R_{max})$$

Substituting these upper bounds in (29) and (28), we obtain

$$|\mathcal{C}^n_{max}| \leq \left(\frac{R_{max}^2}{\tau_{min}}\right)^3 = N_s^3\left(\frac{R_{max}^2}{2\Delta_g^2}\right)^3 = \frac{N_s^3|G|^3}{2^9}$$

$$T_n \leq |G||\mathcal{C}^n_{max}| \leq \frac{N_s^3|G|^4}{2^9}$$

Note that the upper bound derived here is conservative, as can be readily verified through simulations. Figure 16 shows the time and space complexity bounds over $N_s = 8$ iterations for a sample run of the algorithm with $|G| = 1600$ grid points. We observe that in practice, computational complexity increases up to $N_s = 4$ but starts reducing afterwards. The latter is because of the reduction in the number of vacancies in the aperture as the space occupied by the existing subarray modules increases.

ACKNOWLEDGMENT

This work was supported in part by Semiconductor Research Corporation (SRC), DARPA, Center for Scientific Computing at UCSB and National Science Foundation under grants CNS-1518812, CNS-1518632 and CNS-1725797.

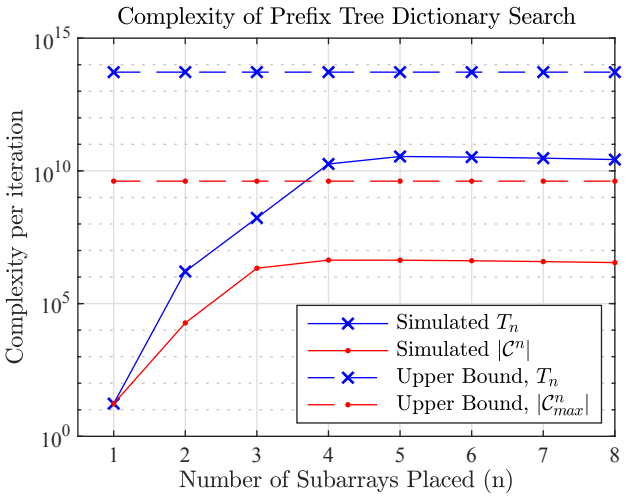


Fig. 16. Complexity over 12 iterations of Prefix Tree Dictionary Search.

REFERENCES

[1] A. Moffet, "Minimum-redundancy linear arrays," *IEEE Transactions on antennas and propagation*, vol. 16, no. 2, pp. 172–175, 1968.

[2] T. Birinci and Y. Tanık, "Optimization of nonuniform planar array geometry for direction of arrival estimation," in *Signal Processing Conference, 2005 13th European*. IEEE, 2005, pp. 1–4.

[3] V. Roy, S. P. Chepuri, and G. Leus, "Sparsity-enforcing sensor selection for doa estimation," in *Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP), 2013 IEEE 5th International Workshop on*. IEEE, 2013, pp. 340–343.

[4] A. Trucco, "Thinning and weighting of large planar arrays by simulated annealing," *IEEE transactions on ultrasonics, ferroelectrics, and frequency control*, vol. 46, no. 2, pp. 347–355, 1999.

[5] P. Pal and P. P. Vaidyanathan, "Nested arrays in two dimensions, part i: Geometrical considerations," *IEEE Transactions on Signal Processing*, vol. 60, no. 9, pp. 4694–4705, Sept 2012.

[6] E. Keto, "Hierarchical configurations for cross-correlation interferometers with many elements," *Journal of Astronomical Instrumentation*, vol. 1, no. 01, p. 1250007, 2012.

[7] S. Joshi and S. Boyd, "Sensor selection via convex optimization," *IEEE Transactions on Signal Processing*, vol. 57, no. 2, pp. 451–462, 2008.

[8] E. Tohidi, M. Coutino, S. P. Chepuri, H. Behroozi, M. M. Nayebi, and G. Leus, "Sparse antenna and pulse placement for colocated mimo radar," *IEEE Transactions on Signal Processing*, vol. 67, no. 3, pp. 579–593, 2018.

[9] P. Jarske, T. Saramaki, S. K. Mitra, and Y. Neuvo, "On properties and design of nonuniformly spaced linear arrays (antennas)," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 36, no. 3, pp. 372–380, 1988.

[10] F. Athley, C. Engdahl, and P. Sunnergren, "On radar detection and direction finding using sparse arrays," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 43, no. 4, 2007.

[11] R. Schmidt, "Multiple emitter location and signal parameter estimation," *IEEE transactions on antennas and propagation*, vol. 34, no. 3, pp. 276–280, 1986.

[12] R. Roy and T. Kailath, "Esprit-estimation of signal parameters via rotational invariance techniques," *IEEE Transactions on acoustics, speech, and signal processing*, vol. 37, no. 7, pp. 984–995, 1989.

[13] K. T. Wong and M. D. Zoltowski, "Direction-finding with sparse rectangular dual-size spatial invariance array," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 34, no. 4, pp. 1320–1336, 1998.

[14] M. D. Zoltowski and K. T. Wong, "Closed-form eigenstructure-based direction finding using arbitrary but identical subarrays on a sparse uniform cartesian array grid," *IEEE Transactions on Signal Processing*, vol. 48, no. 8, pp. 2205–2210, 2000.

[15] M. Haardt and J. A. Nossek, "Simultaneous schur decomposition of several nonsymmetric matrices to achieve automatic pairing in multidimensional harmonic retrieval problems," *IEEE Transactions on Signal Processing*, vol. 46, no. 1, pp. 161–169, 1998.

[16] V. Vasylyshyn and O. Garkusha, "Direction finding using sparse array composed of multiple identical subarrays," in *Antenna Theory and Techniques, 2005. 5th International Conference on*. IEEE, 2005, pp. 273–276.
[17] E. Van den Berg and M. P. Friedlander, "Sparse optimization with least-squares constraints," *SIAM Journal on Optimization*, vol. 21, no. 4, pp. 1201–1229, 2011.
[18] B. Mamandipoor, D. Ramasamy, and U. Madhow, "Newtonized orthogonal matching pursuit: Frequency estimation over the continuum." *IEEE Trans. Signal Processing*, vol. 64, no. 19, pp. 5066–5081, 2016.
[19] C. Stoeckle, J. Munir, A. Mezghani, and J. A. Nossek, "Doa estimation performance and computational complexity of subspace-and compressed sensing-based methods," in *Smart Antennas (WSA 2015); Proceedings of the 19th International ITG Workshop on*. VDE, 2015, pp. 1–6.
[20] D. Ramasamy, S. Venkateswaran, and U. Madhow, "Compressive parameter estimation in awgn." *IEEE Trans. Signal Processing*, vol. 62, no. 8, pp. 2012–2027, 2014.
[21] A. H. Nuttall and B. A. Cray, "Approximations to directivity for linear, planar, and volumetric apertures and arrays," *IEEE journal of oceanic engineering*, vol. 26, no. 3, pp. 383–398, 2001.
[22] K. C. Kerby and J. T. Bernhard, "Sidelobe level and wideband behavior of arrays of random subarrays," *IEEE transactions on antennas and propagation*, vol. 54, no. 8, pp. 2253–2262, 2006.
[23] M. J. Lee, L. Song, S. Yoon, and S. R. Park, "Evaluation of directivity for planar antenna arrays," *IEEE Antennas and Propagation Magazine*, vol. 42, no. 3, pp. 64–67, 2000.
[24] O. Lange and B. Yang, "Antenna geometry optimization for 2d direction-of-arrival estimation for radar imaging," in *Smart Antennas (WSA), 2011 International ITG Workshop on*. IEEE, 2011, pp. 1–8.
[25] K. L. Bell, Y. Ephraim, and H. L. Van Trees, "Explicit ziv-zakai lower bound for bearing estimation," *IEEE Transactions on Signal Processing*, vol. 44, no. 11, pp. 2810–2824, 1996.
[26] J. Proakis and M. Salehi, *Digital Communications, 5th edition*. McGraw-Hill Higher Education, 2008.
[27] Z. Marzi, D. Ramasamy, and U. Madhow, "Compressive channel estimation and tracking for large arrays in mm-wave picocells," *IEEE Journal of Selected Topics in Signal Processing*, vol. 10, no. 3, pp. 514–527, April 2016.
[28] R. A. Horn and C. R. Johnson, *Matrix analysis*. Cambridge university press, 1990.

Anant Gupta received the B.Tech. degree in electronics and electrical communication engineering and the M.Tech. degree in telecommunication systems engineering from IIT Kharagpur in 2013. He received the M.S. degree in electrical and computer engineering from the University of California at Santa Barbara (UCSB) in 2016. He is currently working towards the Ph.D. degree in the Department of Electrical and Computer Engineering, UCSB. His research interests include wireless sensing, statistical signal processing and machine learning.

Upamanyu Madhow is Distinguished Professor of Electrical and Computer Engineering at the University of California, Santa Barbara. His current research interests focus on next generation communication, sensing and inference infrastructures centered around millimeter wave systems, and on robust machine learning. He received his bachelor's degree in electrical engineering from the Indian Institute of Technology, Kanpur, in 1985, and his Ph. D. degree in electrical engineering from the University of Illinois, Urbana-Champaign in 1990. He has worked as a research scientist at Bell Communications Research, Morristown, NJ, and as a faculty at the University of Illinois, Urbana-Champaign. Dr. Madhow is a recipient of the 1996 NSF CAREER award, and co-recipient of the 2012 IEEE Marconi prize paper award in wireless communications. He has served as Associate Editor for the IEEE Transactions on Communications, the IEEE Transactions on Information Theory, and the IEEE Transactions on Information Forensics and Security. He is the author of two textbooks published by Cambridge University Press, *Fundamentals of Digital Communication* (2008) and *Introduction to Communication Systems* (2014).

Amin Arbabian (S'06–M'12–SM'17) received the Ph.D. degree in electrical engineering and computer science from the University of California at Berkeley (UC Berkeley), Berkeley, CA, USA, in 2011. From 2007 and to 2008, he was part of the Initial Engineering Team, Tagarray, Inc., Palo Alto, CA, USA. In 2010, he was with the Qualcomm's Corporate Research and Development Division, San Diego, CA, USA, where he designed circuits for next generation ultralow power wireless transceivers. In 2012, he joined Stanford University, Stanford, CA, USA, as an Assistant Professor of electrical engineering, where he is currently a Frederick E. Terman Fellow with the School of Engineering. His current research interests include high-frequency systems, medical imaging, Internet-of Everything devices including wireless power delivery techniques, and medical implants. Dr. Arbabian was a recipient or a co-recipient of the 2015 NSF CAREER Award, the 2014 DARPA Young Faculty Award, the 2013 IEEE International Conference on Ultra-Wideband Best Paper Award, the 2013 Hellman Faculty Scholarship, the 2010 IEEE Jack Kilby Award for Outstanding Student Paper of the International Solid-State Circuits Conference, two-time Second Place Best Student Paper Award at the 2008 and 2011 Radio-Frequency Integrated Circuits (RFIC) Symposium, the 2009 Center for Information Technology Research in the Interest of Society at UC Berkeley Big Ideas Challenge Award and the UC Berkeley Bears Breaking Boundaries Award, and the 2010-2011 and 2014-2015 Qualcomm Innovation Fellowship. He currently serves on the TPC for the European Solid-State Circuits Conference and the RFIC Symposium.

Ali Sadri is Sr. Director of mmWave Standards and Advanced Technologies at Intel Corporation. His Professional work started at IBM in 1990 as Director of wireless communications standards and a also a visiting assistant Professor at the Duke University through 2000. Upon joining Intel Corporation, Dr. Sadri initiated and lead the standardization of the IEEE 802.11n standard. In 2008 Ali founded and led the Wireless Gigabit Alliance to create the ground breaking WiGig 60 GHz technology and later merged with WiFi Alliance In June 2013 to advance and certify the WiGig programs within WiFi alliance framework. Currently Ali is leading the mmWave standards and advanced technology development to enable 5G mmWave systems. Ali holds more than 100 Issued and pending patent applications in wired and wireless communications systems.