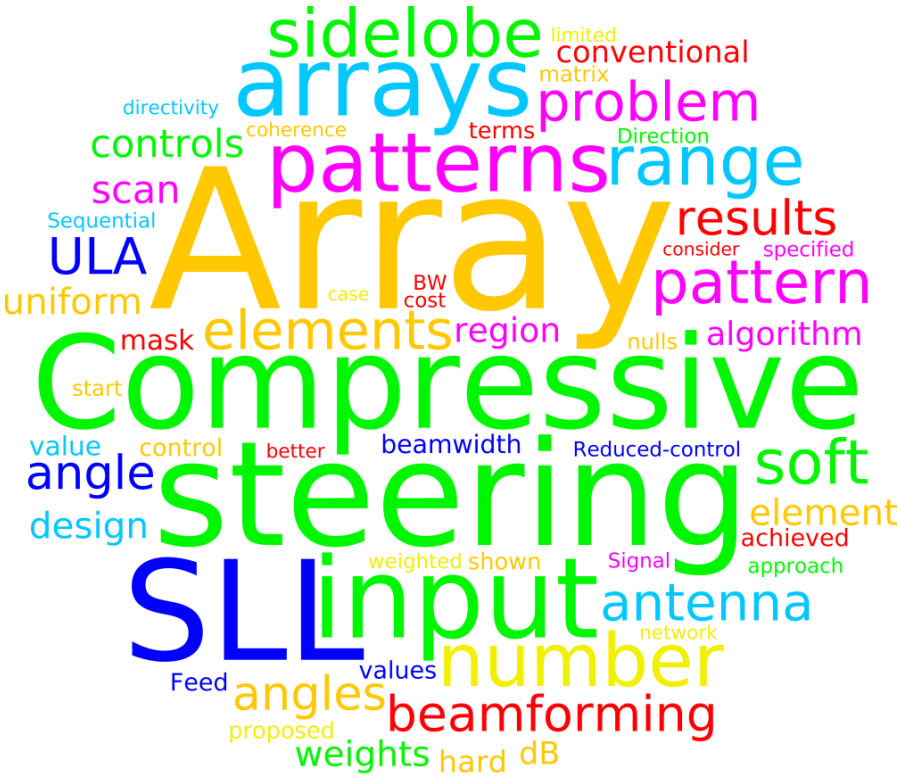
Accepted version of: Heinrich Edgar Arnold Laue and Warren Paul du Plessis, “Numerical Optimisation of Compressive Array Feed Networks,” *IEEE Transactions on Antennas and Propagation*, accepted on 1 April 2018. DOI: 10.1109/TAP.2018.2829834



ABBREVIATIONS

CS	compressive sensing
CSA	compressive sampling array
CSIR	Council for Scientific and Industrial Research
DF	direction-finding
IFT	iterative FFT technique
MVDR	minimum variance distortionless response
NRF	National Research Foundation of South Africa
SLL	sidelobe level
SQP	sequential quadratic programming
UCA	uniform circular array
ULA	uniform linear array

Numerical Optimisation of Compressive Array Feed Networks

Heinrich Edgar Arnold Laue, *Student Member, IEEE,* and Warren Paul du Plessis, *Senior Member, IEEE*

***Abstract*—Compressive antenna arrays reduce the number of beamforming controls by taking a limited number of weighted combinations of the element signals. A compressive array sidelobe level (SLL)-minimisation algorithm, inspired by coherence-optimisation algorithms, is presented. Optimised compressive arrays are shown to obtain better SLLs than existing weighted thinned arrays and completely overlapped subarrays. The design of a compressive array with arbitrary sidelobe requirements shows that increasing the number of elements for a given number of beamforming controls improves array performance over a conventional array. A compressive array with a hard null is proposed to suppress interference before sampling. Where beamforming controls are the main cost drivers, the proposed approach promises to increase array performance without a significant increase in cost.**

***Index Terms*—Antenna arrays, antenna feeds, antenna pattern synthesis, coherence, compressive sensing (CS), phased arrays.**

I. Introduction

BEAMFORMING antenna arrays utilise multiple antenna elements in order to steer a beam in a particular direction while the array itself remains stationary [1]. In traditional phased arrays, each element is connected to an analogue phase shifter, and possibly, an amplifier [2]. On the other hand, element-level digital beamforming arrays utilise a transmitter and/or receiver at each antenna element [2]. These hardware components will be referred to as beamforming controls since they enable adaptive beamforming. Advantages of digital beamforming include the ability to steer multiple beams simultaneously in software, array reconfigurability, improved dynamic range, and precise array calibration [1]–[3].

The use of phase shifters and/or amplifiers at each element in traditional phased arrays, and transmitters and/or receivers at each element in digital antenna arrays places limitations on the number of elements that may be used due to the size, weight, complexity, and cost involved [2]–[4]. With uniform spacing between antenna elements, this limits the aperture of the array, and therefore, the achievable resolution.

Over the decades, significant effort has gone into developing arrays with a given aperture that require either fewer antenna elements, or fewer beamforming controls in an effort to reduce cost, size and weight [2], [4]–[6]. Any array that aims to reduce the number of beamforming controls will be referred to as a reduced-control array below.

One way of reducing the cost and complexity of an array is to take a filled array and disable some of the elements, referred to as array thinning [6]–[8]. The result is an array with a beamwidth similar to that of the filled array, but with decreased directivity proportional to the number of active elements [1], [6]. The remaining elements may also be weighted for finer control over the array patterns [6], [9]. A similar approach where the antenna elements may be arbitrarily positioned within an array leads to sparse arrays [5], [10].

Another reduced-control technique is to combine the signals at the antenna elements before reaching the beamforming controls, thereby reducing the number of controls for the same number of antenna elements [1], [2], [4], [11]. Subarrays combine antenna elements for a reduced steering range, resulting in narrow beams with high directivity being steered over a relatively small angular range [1], [2], [4].

An array which combines the antenna-element signals so that each output is a function of all antenna-element signals was inspired by the concept of compressive sensing (CS) for direction-finding (DF) and is termed the compressive sampling array (CSA) [11]. A CSA can steer across the full field of view. Initially, the use of random weights to combine the element signals was proposed [11]. Subsequently, the use of numerically optimised codebooks was shown to improve on the use of random weights, but only allowed control over the array pattern at a small number of angles [12].

A generalised framework for the numerical optimisation of compressive feed networks for narrowband pencil-beam arrays with arbitrary sidelobe-pattern requirements is presented. This framework includes other reduced-control networks as special cases. The framework is inspired by coherence-optimisation algorithms which optimise codewords to have minimal similarity, or coherence [13]. The coherence-optimisation problem is extended to consider the problem of minimising sidelobe level (SLL) in a compressive array. The presented approach allows the sidelobe amplitude to be specified independently of pattern and steering angle, which is useful as shaped sidelobe patterns are often desirable in antenna arrays [14].

Manuscript received 23 September 2017; revised 6 February 2018; accepted 1 April 2018.

This work is based on the research supported in part by the National Research Foundation of South Africa (NRF) (Grant specific unique reference number (UID) 85845). The NRF Grantholder acknowledges that opinions, findings and conclusions or recommendations expressed in any publication generated by the NRF supported research are that of the author(s), and that the NRF accepts no liability whatsoever in this regard.

The authors are with the Department of Electrical, Electronic and Computer Engineering, University of Pretoria, Pretoria 0002, South Africa (e-mail: laueheinrich@gmail.com; wduplessis@ieee.org)

Color versions of one or more of the figures, as well as MATLAB scripts for generating the main results in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 00.0000/TAP.0000.000000

The proposed compressive-array framework is not restricted to a particular hardware configuration, but represents a flexible reduced-control array design methodology. For example, a compressive feed network may be implemented either using microwave circuitry to enable an array to have a larger number of antenna elements than beamforming controls, or as a software algorithm to reduce data rates.

The versatility of the proposed algorithm is illustrated by synthesising linear and circular compressive arrays with a variety of constraints, including hard and soft nulls. Comparisons to optimal solutions [15], thinned arrays synthesised using the iterative FFT technique (IFT) [9], conventional arrays synthesised by the matrix inversion method [14], and a dual-transform completely overlapped subarray [1] demonstrate the performance of compressive arrays synthesised using the proposed approach.

Section II provides the required background, describes existing methods for designing compressive arrays, and gives an overview of discrete coherence optimisation. Section III formulates the continuous coherence-optimisation problem for minimising SLL and presents an algorithm for accomplishing this goal. Section IV presents results for a number of designs which highlight various advantages of the proposed approach. Finally, conclusions are drawn in Section V.

II. Background

A. Conventional Antenna Arrays

Consider an array of N antenna elements placed at arbitrary locations in two-dimensional space. Define the steering vector as the baseband array response to a unit-amplitude continuous-wave signal impinging on the array from the direction θ in the plane of the array, denoted $\mathbf{a}(\theta)$.[1] The steering vector includes the effect of the element patterns and can either be formulated mathematically or measured practically. In the case of a uniform linear array (ULA) with isotropic elements, the $N\times 1$ steering vector is given by [16]

$$\mathbf{a}(\theta) = \left[\begin{array}{ccc} e^{j\beta d_1 \sin(\theta)} & \cdots & e^{j\beta d_N \sin(\theta)} \end{array} \right]^{\mathrm{T}} \qquad (1)$$

where θ is the anticlockwise angle of the wave relative to broadside, d_n is distance of element n to a reference point on the array, and $\beta = 2\pi/\lambda$, with λ the wavelength.[2] A circular array with radius R and isotropic elements on the circumference of the circle has a steering vector given by [16]

$$\mathbf{a}(\theta) = \left[\begin{array}{ccc} e^{j\beta R \sin(\theta-\theta_1)} & \cdots & e^{j\beta R \sin(\theta-\theta_N)} \end{array} \right]^{\mathrm{T}} \qquad (2)$$

where θ_n is the angle of element n around the circle. The steering of circular arrays will only be considered in the plane of the array. In subsequent sections, the steering vectors are normalised to have unit length to maintain proper scaling of constraints on the steering vectors.

[1]The techniques described are applicable in the general case with suitable alterations, but only the one-dimensional case is considered to avoid unnecessary notational complexity.

[2]It is worth noting that since the analyses are phase-based, the results are inherently narrowband. However, most antenna-array analyses are formulated in this way, so this is not considered a limitation of the results obtained.

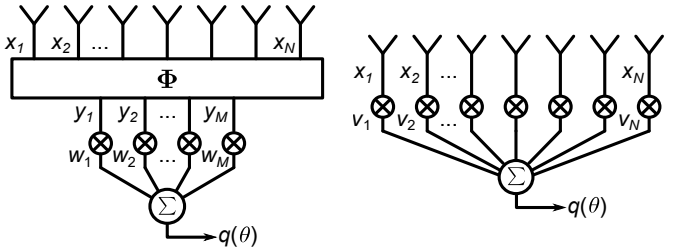


Fig. 1. (a) A compressive array with N elements and M subarray outputs with digital beamforming weights for array pattern calculations. (b) An N-element ULA with weights that result in the same patterns as the corresponding compressive array.

B. Compressive Arrays

Consider the general compressive array illustrated in Fig. 1(a), shown in the receive configuration. The results obtained are, however, equally valid for a transmit antenna due to reciprocity. The system is represented by $\mathbf{y} = \mathbf{\Phi}\mathbf{x}$, where $\mathbf{y}$ is the $M\times 1$ vector of signals at the beamforming controls, $\mathbf{x}$ is the $N\times 1$ vector of signals at the antenna elements, and $\mathbf{\Phi}$ is the $M \times N$ sensing matrix representing a compressive feed network with $M \leq N$ [12]. The rows of $\mathbf{\Phi}$ describe how the N elements are weighted to form each of the M subarray outputs. The $M\times 1$ compressed steering vector (the response $\mathbf{y}$ to a reference wave) is then $\mathbf{b}(\theta) = \mathbf{\Phi}\mathbf{a}(\theta)$ [11], [12].

A conventional uniform-excitation array with one element per beamforming control can be represented by the $N \times N$ sensing matrix $\mathbf{\Phi} = \mathbf{I}$. A non-weighted thinned array will have an $M \times N$ sensing matrix made up of M rows taken from an $N \times N$ identity matrix [11]. A partially overlapped subarray [4] will have a certain number of non-zero entries per row, while a completely overlapped subarray [1] can be represented by a fully populated sensing matrix. The CSA in [11] uses a sensing matrix with random Gaussian entries.

The term compressive array arises as the synthesis of antenna arrays with complex-valued sensing matrices with $M < N$ is considered, similar to the approach used in CS [17], [18]. As outlined above, the compressive-array formulation includes most existing reduced-control arrays with fixed element positions as special cases, making it a generalisation of the reduced-control concept. By using this generalisation, it is shown how sensing matrices may be designed to improve on existing criteria and fulfil a variety of criteria that could not previously be considered for reduced-control arrays.

A compressive feed network can be implemented in hardware or in software, for transmission and/or reception. Overlapped subarrays suggest that feed networks can be implemented in microwave circuity where each output is a weighted combination of multiple antenna elements [1], [2], [4], [19]. The proposed method could also be applied to an array with a receiver at each element by using a software implementation of the sensing matrix. This would enable reduced data rates for transmission to a central processing station, for example.

C. Compressive Array Patterns

A set of M complex-valued excitations at the M subarray outputs is defined to characterise a compressive array. The feed network, described by $\mathbf{\Phi}$, results in compressed steering vectors $\mathbf{b}(\theta)$ which change in both amplitude and phase with steering angle. This requires each steering angle to have its own set of weights to achieve the desired pattern when steering in that direction.

Considering the illustration in Fig. 1(a), the array voltage pattern is defined as the weighted sum of the M subarray outputs due to a reference wave from the direction θ as

$$q(\theta) = \left|\mathbf{w}(\theta_s)^{\mathrm{T}}\mathbf{b}(\theta)\right| \tag{3}$$

where $\mathbf{w}(\theta_s)$ is the $M\times 1$ vector of complex weights for the steering angle θ_s.

The aim of a compressive array design algorithm is to find $\mathbf{\Phi}$ and $\mathbf{w}(\theta_s)$ for all steering angles of interest that minimise the SLL. The SLL is defined as the largest pattern magnitude in a pre-defined sidelobe region relative to the main-beam peak. Since $\mathbf{b}(\theta)$ is a function of $\mathbf{\Phi}$, both $\mathbf{w}(\theta_s)$ and $\mathbf{b}(\theta)$ must be optimised over all steering angles θ_s and pattern angles θ. However, the problem can be greatly simplified if $\mathbf{w}(\theta_s)$ is chosen as the complex conjugate of $\mathbf{b}(\theta_s)$. This leads to

$$q(\theta) = \left|\mathbf{b}(\theta_s)^{\mathrm{H}}\mathbf{b}(\theta)\right| \tag{4}$$

where $^{\mathrm{H}}$ represents the complex-conjugate transpose. This expression is similar in form to the well-known coherence criterion from CS theory, which will be discussed in Section II-D.

In order to validate the compressive array algorithm for an $M = N$ array against a Chebyshev array (Section IV-A), it is necessary to define a ULA with the same patterns as the corresponding compressive array. To do this, consider the combined effect of the sensing matrix $\mathbf{\Phi}$ and the beamforming weights $\mathbf{w}(\theta_s)$. Substituting $\mathbf{b}(\theta) = \mathbf{\Phi}\mathbf{a}(\theta)$ into (4) gives

$$q(\theta) = \left|\mathbf{a}^{\mathrm{H}}(\theta_s)\mathbf{\Phi}^{\mathrm{H}}\mathbf{\Phi}\mathbf{a}(\theta)\right|. \tag{5}$$

Since $\mathbf{a}(\theta)$ is the steering vector for a ULA, define a set of equivalent weights for a conventional ULA as $\mathbf{v}^{\mathrm{T}}(\theta_s) = \mathbf{a}^{\mathrm{H}}(\theta_s)\mathbf{\Phi}^{\mathrm{H}}\mathbf{\Phi}$, or

$$\mathbf{v}(\theta_s) = \mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}^{*}\mathbf{a}^{*}(\theta_s) \tag{6}$$

where * represents complex conjugation. This gives

$$q(\theta) = \left|\mathbf{v}(\theta_s)^{\mathrm{T}}\mathbf{a}(\theta)\right| \tag{7}$$

which is similar in form to (3). Applying the weights $\mathbf{v}(\theta_s)$ to a length-N ULA, as illustated in Fig. 1(b), will result in the same patterns as the corresponding compressive array.

While varying excitations is the only way to control transmit beamforming, digital beamforming techniques on reception are not limited to weight-and-sum beamforming. For example, super-resolution methods such as minimum variance distortionless response (MVDR) and CS-based algorithms have been successfully applied to compressive arrays for DF [11], [12].

D. Discrete Coherence Optimisation

Consider N length-M vectors as columns in the codebook $\mathbf{B}$. The coherence of a codebook quantifies how closely the vectors approximate an orthogonal set and is given by [13]

$$\mu(\mathbf{B}) = \max_{n\neq l}\frac{\left|\mathbf{b}_n^{\mathrm{H}}\mathbf{b}_l\right|}{\|\mathbf{b}_n\|\,\|\mathbf{b}_l\|}, \quad n,l\in\{1,\ldots,N\}. \tag{8}$$

Often, the codewords are assumed to be normalised, in which case the coherence simplifies to $\max_{n\neq l}\left|\mathbf{b}_n^{\mathrm{H}}\mathbf{b}_l\right|$ [20].

Optimising the coherence of a codebook is given by the minimax problem [13], [21]

$$\min_{\mathbf{B}}\ \max_{n\neq l}\left|\mathbf{b}_n^{\mathrm{H}}\mathbf{b}_l\right| \ \text{ subject to } \ \|\mathbf{b}_n\|^2 = 1 \quad \forall\, n. \tag{9}$$

Various methods for designing codebooks with minimal coherence exist [13], [21]–[24]. The best results in terms of coherence and computational efficiency were obtained in [13], which solved a series of subproblems given by

$$\min_{\mathbf{B}}\sum_{n\neq l}\left(\left|\mathbf{b}_n^{\mathrm{H}}\mathbf{b}_l\right|^2 - \mu_t^2\right)^p \quad p = 2,4,\ldots$$
$$\text{subject to } \ \|\mathbf{b}_n\|^2 = 1 \quad \forall\, n \tag{10}$$

which minimises the absolute deviation of the pairwise absolute dot products from some target bound μ_t, which is derived from the known minimum possible value. The solution to the subproblem with the first value of p is used as starting point for the next subproblem. Larger values of p assign more weight to the larger summation terms, thereby providing an increasingly-accurate approximation to the $\max$ operator as $p\to\infty$.

During the initial stages when the parameters change significantly, small values of p provide the good conditioning required for numerical stability [22]. Incrementing p guides the problem towards an accurate approximation of the $\max$ operator while maintaining numerical stability, even though a large p would represent an ill-conditioned problem if the starting point was not already close to a local minimum.

A codebook $\mathbf{B}$ with optimised coherence can be used to design a sensing matrix for a compressive array via [12]

$$\mathbf{\Phi} = \mathbf{B}\mathbf{A}^{-1} \tag{11}$$

where $\mathbf{A}$ is the conventional steering matrix $\mathbf{A} = [\,\mathbf{a}(\theta_1)\ \cdots\ \mathbf{a}(\theta_N)\,]$ at the N steering angles at which the steering vectors are orthogonal. Such an optimised array has been applied for DF and shown to perform better than compressive arrays with random Gaussian sensing matrices [12]. However, it was noted that this method only allows for sidelobe control at N discrete angles in the array patterns [12].

III. Continuous SLL Optimisation

A. Problem Formulation

The use of optimised codebooks to design compressive arrays for low SLL is severely hampered by the fact that array pattern control is only possible at N angles. Even if the discrete $M\times N$ codebook $\mathbf{B}$ has the lowest possible coherence, there is no way to predict the resulting array patterns between the N angles. Simply increasing the number of sampling

angles in $\mathbf{A}$ will not suffice since this would result in $\mathbf{B} = \mathbf{\Phi A}$ from (11) being under-determined.

In order to exercise sidelobe control across continuous steering and pattern angles, it is necessary to reformulate the optimisation problem in terms of $\mathbf{\Phi}$, since the sensing matrix with MN discrete complex weights determines the continuous compressed steering vectors via $\mathbf{b}(\theta) = \mathbf{\Phi a}(\theta)$.

With the definition of the excitation weights as the complex conjugates of the compressed steering vectors (Section II-C), the SLL can be computed from

$$\begin{aligned} \mathrm{SLL} = \max_{|\theta_a - \theta_s| \geq \theta_{\mathrm{SLL}}} \left| \mathbf{b}(\theta_s)^{\mathrm{H}} \mathbf{b}(\theta_a) \right|, \\ \theta_s \in [\theta_{s1}, \theta_{s2}], \quad \theta_a \in [\theta_{a1}, \theta_{a2}] \end{aligned} \tag{12}$$

where θ_{SLL} is the specified start of the sidelobe region relative to the steering angle, θ_s are the steering angles, θ_a are the array pattern angles, $[\theta_{s1}, \theta_{s2}]$ is the steering range, and $[\theta_{a1}, \theta_{a2}]$ is the field of view (typically, $[-90°, 90°]$ for ULAs and $[-180°, 180°]$ for circular arrays). When designing compressive feed networks for ULAs, the start of the sidelobe region is defined to be constant in $\sin(\theta)$ space, with the result that it changes with steering angle (θ_s) in θ space.

Note that $\mathbf{b}(\theta_s)^{\mathrm{H}} \mathbf{b}(\theta_a) = \left[\mathbf{b}(\theta_a)^{\mathrm{H}} \mathbf{b}(\theta_s) \right]^*$ and therefore $\left| \mathbf{b}(\theta_s)^{\mathrm{H}} \mathbf{b}(\theta_a) \right| = \left| \mathbf{b}(\theta_a)^{\mathrm{H}} \mathbf{b}(\theta_s) \right|$, which explains the symmetry observed in the pattern plots of Section IV. Thus only unique combinations of θ_s and θ_a need to be considered in the optimisation, which significantly reduces the problem complexity. Unique combinations of θ_s and θ_a are given by $\theta_a - \theta_s \geq \theta_{\mathrm{SLL}}$ (the upper triangular sidelobe regions in Figs. 3 and 4 in Section IV).

Define the continuous coherence-optimisation problem as

$$\begin{aligned} \min_{\mathbf{\Phi}} \max_{\theta_a - \theta_s \geq \theta_{\mathrm{SLL}}} \frac{\left| [\mathbf{\Phi a}(\theta_s)]^{\mathrm{H}} [\mathbf{\Phi a}(\theta_a)] \right|}{R(\theta_s, \theta_a)} \\ \text{subject to} \\ \left\| \mathbf{\Phi a}(\theta_s) \right\|^2 = 1, \quad \theta_s \in [\theta_{s1}, \theta_{s2}], \quad \theta_a \in [\theta_{a1}, \theta_{a2}] \end{aligned} \tag{13}$$

where $R(\theta_s, \theta_a)$ is the sidelobe mask. This mask is used to allow the sidelobe requirements to vary over the sidelobe region. Note that the mask values may be specified independently for both steering and pattern angles, which leads to a number of novel designs as discussed in Section IV.

The condition $\left\| \mathbf{b}(\theta_s) \right\|^2 = 1$ is necessary to ensure that a peak is located at θ_s when the beam is steered in that direction. In the algorithm to follow, this constraint is relaxed so that the magnitudes of the compressed steering vectors are between ± 0.01 dB because this relaxation leads to faster convergence.

Similar to the discrete coherence-optimisation problem in (9), the continuous formulation in (13) is non-convex and is not guaranteed to converge to the global minimum. Even so, local optimisers produce discrete codebooks with coherence values that have not been improved upon by any other method [13], [21], [22], so this approach is reasonable.

Unfortunately, optimising the patterns using the formulation above would involve evaluating an infinite number of combinations of steering and pattern angles. The angles are thus discretised to provide a finite set of angles over which the SLL must be optimised. This is done by defining a set of FN sampling angles for both the steering and pattern angles, where F is the oversampling factor. These angular points are uniformly spaced in $\sin(\theta)$ space over $[-1,1)$ for ULAs to compensate for the $\sin(\theta)$ factor in (1), and in θ space over $[-180°, 180°)$ for circular arrays as the elements are uniformly distributed around the circle. Although FN sampling angles are defined, only those within the steering range and sidelobe regions are considered in the optimisation.

Since the pattern values must be evaluated over all relevant sampling angles, the problem is inherently combinatorial in nature, with up to $\mathcal{O}\left\{ \binom{FN}{2} \right\}$ combinations to consider. The oversampling factor F allows a compromise to be made between angular grid resolution (which influences the achieved SLL) and realistic run times. A value of $F = 2M$ was empirically found to present a good compromise between achieved SLL and run time. Using $F = 2M$ means that the resulting number of sampling angles, $2MN$, is equal to the number of control variables, namely the real and imaginary parts of the MN elements in $\mathbf{\Phi}$.

In order to ensure that the calculated performance is accurate the final beamwidth and SLL values in the results below were calculated using 16 times more sampling points [9]. The 3-dB points in the beamwidth calculations were found using cubic spline interpolation [9].

B. Algorithm and Implementation

The approach to minimising SLL in terms of $\mathbf{\Phi}$ is to sequentially approximate the $\max$ operator similar to (10) and use a general-purpose constrained non-linear solver for the subproblems. The goal function to be minimised is given by

$$g(\mathbf{\Phi}) = \alpha \left[\frac{1}{C} \sum_{\substack{\theta_a - \theta_s \geq \\ \theta_{\mathrm{SLL}}}} \left(\frac{\left| (\mathbf{\Phi a}_s)^{\mathrm{H}} (\mathbf{\Phi a}_a) \right|}{\alpha R_{s,a}} \right)^p \right]^{1/p} \tag{14}$$

subject to the constraints

$$c_1(\mathbf{\Phi}) = \left\| \mathbf{\Phi a}_s \right\|^2 - 10^{0.01/10} \leq 0 \quad \forall s \quad \text{and} \tag{15}$$

$$c_2(\mathbf{\Phi}) = 10^{-0.01/10} - \left\| \mathbf{\Phi a}_s \right\|^2 \leq 0 \quad \forall s \tag{16}$$

where the subscripts s and a refer to variables sampled at the angles θ_s and θ_a, and C is the number of terms in the summation. Including the factor $1/C$ and raising the sum to $1/p$ maintains proper scaling of the problem. The function is of the form $\left(\sum_N |x_n|^p \right)^{1/p}$ which approximates the ∞–norm by the p–norm as $p \to \infty$ [22]. The inequality-constraint functions for the unit-length requirement in (13) are provided in (15) and (16).

Since the dot products scaled by the sidelobe mask are typically smaller than one, raising them to large powers may produce results smaller than the lower limit of numerical representation, resulting in underflow of some of the terms in the summation. A scaling factor α, which does not affect the final function value, is thus introduced to scale the terms in the summation to minimise the incidences of underflow. The terms in the summation in (14) can be written as $(x_{s,a}/\alpha)^p$, and the goal is to ensure that these terms are as large as possible.

Overflow during the intermediate computations can be avoided by setting the sum equal to the largest representable value and assuming that all terms are equal to the largest term, giving

$$\sum_{C} \left(\frac{x_{\max}}{\alpha}\right)^p = C\left(\frac{x_{\max}}{\alpha}\right)^p = \nu_{\max} \tag{17}$$

where $\nu_{\max}$ is the largest representable number. Solving for α gives

$$\alpha = \frac{x_{\max}}{(\nu_{\max}/C)^{1/p}}. \tag{18}$$

The sequential quadratic programming (SQP) algorithm [25] in MATLAB was chosen to solve the subproblems given by (14) to (16) for increasing values of p. The value of p is incremented via

$$p^{(k)} = \min\left(p_{\max},\ 2\left\lceil \frac{rp^{(k-1)}}{2} \right\rceil\right), \quad k = 2,3,\ldots,\quad p^{(1)} = 2,\quad r > 1 \tag{19}$$

where r is the power multiplication factor, k is the subproblem number, and $p_{\max}$ is the maximum value of p. For each subproblem, p is incremented to the next even number greater or equal to $rp^{(k-1)}$. Restricting p to even values ensures that (14) is smooth [13]. For a particular problem, the value of $p_{\max}$ should be increased until no further improvement in SLL is observed, after which the value of r is then reduced until no further improvement in SLL is observed. Values of $p_{\max} = 512$ [13] and $r = 1.1$ have been found to produce good solutions for a wide range of problems and were used to obtain the results in Section IV.

The initial sensing matrix $\mathbf{\Phi}^{(0)}$ was chosen with real and imaginary parts drawn from a Gaussian distribution with variance $1/(2N)$ so that $\mathbb{E}\left[\|\boldsymbol{\phi}_m\|\right] = 1$, where $\boldsymbol{\phi}_m$ are the rows in $\mathbf{\Phi}$ [26]. The minimum step size was set to 10^{-10}, the step size used for discrete coherence optimisation in [13]. The first-order optimality and constraint convergence criteria were kept at their default values of 10^{-6}. The number of iterations per sub-problem was limited to a maximum of 10^5 [13]. Gradients of goal and constraint functions were derived analytically and are given in the appendix. This is important since using finite differences is computationally inefficient.

IV. Results

The results obtained for a number of test problems are outlined below, after a description of the conditions under which the results were obtained.

Isotropic antenna elements are assumed. However, the procedure is general enough that the steering vectors may be specified in terms of arbitrary element patterns including, for example, simulated or measured embedded element patterns.

Angles and beamwidths for ULAs are given in $u = \sin(\theta)$ space. The algorithm was implemented in MATLAB R2016b and run on machines with two 6-core 2.30-GHz Intel Xeon E5-2630 processors and 32 GB of memory each. In all cases, an oversampling factor of $F = 2M$ was used.

The results below are summarised in Table I. The SLL and beamwidth results for the compressive arrays are for the designs with the lowest SLL from ten independent runs of the algorithm. Only the worst (i.e. largest) 3-dB beamwidths over the steering ranges are presented. The run times indicated are for the algorithm to run to $p = p_{\max}$ even if the best SLL was obtained before then.

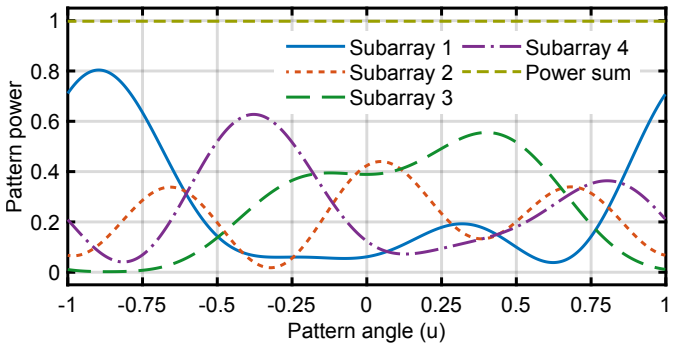


Fig. 2. Subarray power patterns and their sum for the $M = N = 4$ compressive array in Section IV-A.

Reduced-control arrays can be considered to either reduce the number of beamforming controls for a given aperture or to increase the aperture for a given number of beamforming controls. Under the reasonable assumption that the cost of an array is primarily determined by the number of beamforming controls [2], the second perspective is more useful when comparing reduced-control and conventional arrays for a given cost. Section IV-C below thus compares a compressive array with M controls and N antenna elements to an M-element conventional array because an N-element conventional array would cost approximately N/M more.

A. $M = N$, Uniform Sidelobe Mask

The first test case designs a feed network with $M = 4$ outputs for a ULA with $N = 4$ elements, the start of the sidelobe region at $u_{\text{SLL}} = 0.5$, and a uniform sidelobe mask. Since $M = N$, a Chebyshev excitation provides the optimum result for the conventional weights $\mathbf{v}(\theta_s)$, where optimality refers to the lowest possible SLL for a given main beam region, or the smallest beamwidth for a given SLL [15]. This problem thus serves to test whether the proposed algorithm is able to approach the known optimum result.

For the purpose of validation, the steering range was defined as $u_s \in [-1,\ 1)$ since a ULA has patterns which translate all the way to endfire, though with a grating lobe at endfire when half-wavelength spacing is used. When calculating SLL, the sidelobe region range was adjusted to ensure that that grating lobes were ignored as is done for Chebyshev arrays.

The compressive array achieved an SLL of -16.98 dB. The Chebyshev excitation for an SLL of -16.98 dB is $|\mathbf{v}(\theta_s)| = [0.6669,1,1,0.6669]^{\mathrm{T}}$. The normalised magnitudes of the equivalent weights for the compressive array are the same, with a maximum deviation of 9.89×10^{-7} from the Chebyshev excitation over all weights and steering angles. This result demonstrates that the compressive array algorithm can achieve comparable results to the known optimum.

Fig. 2 shows the antenna patterns of each of the M subarrays comprising the compressive array design. The patterns are

TABLE I
RESULTS FOR COMPRESSIVE ARRAY DESIGNS OVER TEN RUNS WITH COMPARISONS TO EXISTING APPROACHES

	Sequential compressive array algorithm				Comparison		
Problem (Section)	SLL (dB)	$\text{BW}_{3\,\text{dB}}$	Min. run time (s)	Max. run time (s)	Approach	SLL (dB)	$\text{BW}_{3\,\text{dB}}$ (u)
$M = N$ (IV-A)	-16.98	0.4994 (u)	10.4	16.3	Chebyshev	-16.98	0.4994
Uniform mask (IV-B)	-2.89	0.0942 (u)	4 553.8	14 497.3	IFT [9]	-2.47	0.1017
Soft null (IV-C)	-19.85	0.2370 (u)	448.5	11 610.5	Length-M ULA [14]	-19.76	0.2556
Hard null (IV-D)	-13.27	50.1°	14 849.2	27 296.4	—		
Small steering range (IV-E)	-34.20	0.0382 (u)	3 657.7	3 806.1	Dual-transform [1]	-32.49	0.0398

shown as power values, which add to 1 at each angle as shown and as required by (13). The subarray patterns are simply the M individual elements in the compressed steering vector, since each element in $\mathbf{b}(\theta)$ represents the response of the array at the mth beamforming control. Even though the excitations at the N elements are equivalent to those of a Chebyshev array, the operating principle is different. A weighted ULA with M isotropic elements samples the entire angular domain at each element. In the compressive array, each subarray samples the angular space differently. The subarrays are able to scan the entire range of interest when combined, but individually, some subarrays favour certain angles. The crucial observation is that there is no angle at which all of the subarray gains are low because that would make steering in that direction impossible.

The ability of compessive arrays to implement shaped subarray patterns can be exploited to suppress the subarray patterns where no steering is desired (Section IV-C), or to suppress interference from a particular direction (Section IV-D).

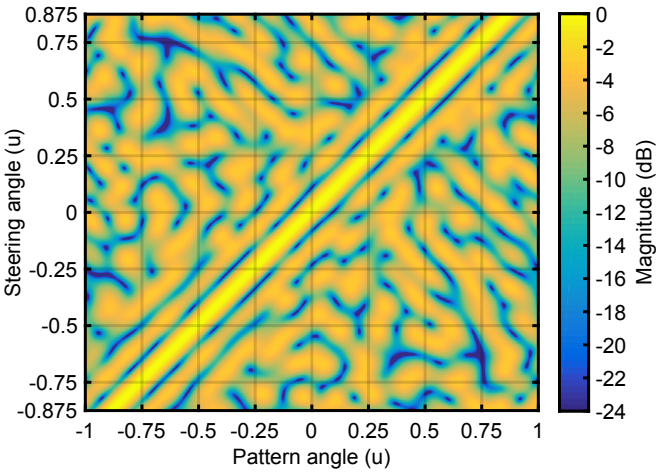


Fig. 3. Normalised patterns as a function of steering angle for the uniform-mask compressive array problem in Section IV-B.

B. $M < N$, Uniform Sidelobe Mask

The true compressive array design evaluated here considers a feed network with $M = 4$ for a ULA with $N = 16$, the start of the sidelobe region at $u_{\text{SLL}} = 0.046875$, and a uniform sidelobe specification. The steering range was limited to $|u_s| \leq 0.875$ or $|\theta_s| \leq 61°$ since $|\theta_s| \leq 60°$ is already considered a wide steering range and steering to endfire presents a number of practical challenges [1]. This problem illustrates the case where beamwidths similar to that of a large, filled array are desired at the expense of higher SLL, a typical consideration for DF arrays.

The results are compared to a weighted thinned linear array designed using the IFT [6], [9]. The IFT does not allow the steering range to be limited, so this example also serves to illustrate the performance improvements arising from the ability of compressive arrays to limit the steering range.

Fig. 3 shows the patterns as a function of steering angle for the compressive array design. Significantly, the patterns do not translate with steering angle, but vary significantly as the main beam is steered. Note that beamwidth and SLL values provided below are achieved at all steering angles despite the significant pattern variation. This is an example of non-translational patterns which arise in compressive arrays. One of the main consequences of non-translational patterns is that steering of the main beam is more complex as the excitation amplitudes and phases vary non-linearly, unlike translational patterns where only linear phase variation is required.

The compressive array has an SLL of -2.89 dB and a worst-case 3-dB beamwidth of 0.0942 over the specified steering range, while the IFT achieved an SLL of -2.47 dB with a 3-dB beamwidth of 0.1017 as the best result from ten runs. The compressive array design improves on the weighted thinned design by 0.42 dB in SLL and 7.4% in beamwidth (in u). This improvement is achieved by exploiting the additional degrees of freedom provided by the fact that the M feed network outputs are functions of all N antenna elements, which includes the ability to exploit a limited steering range. On the other hand, in a weighted thinned array each of the M outputs is only a function of a single antenna, and the IFT cannot exploit a limited steering range.

Compressive arrays with uniform sidelobe masks would be useful as an alternative to thinned or weighted thinned arrays where SLL is crucial, since the additional degrees of freedom in the feed network allow for designs with lower SLL for the same number of beamforming controls.

C. Soft Stationary Null Steering

Section IV-B showed that a compressive array can have patterns that change as a function of steering angle. In order to exploit this property, the sidelobe mask $R_{s,a}$ can be specified

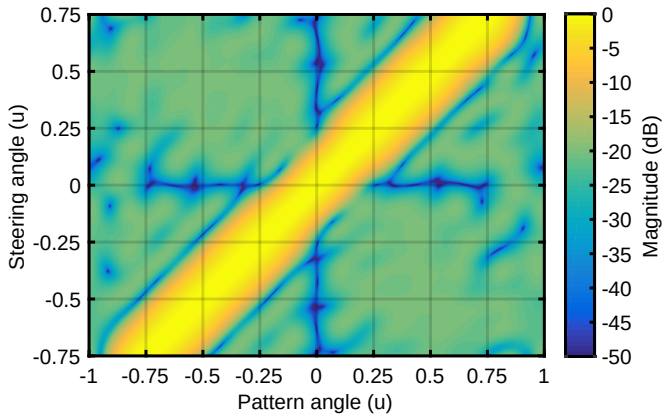


Fig. 4. Normalised patterns as a function of steering angle for the soft-null compressive array problem in Section IV-C.

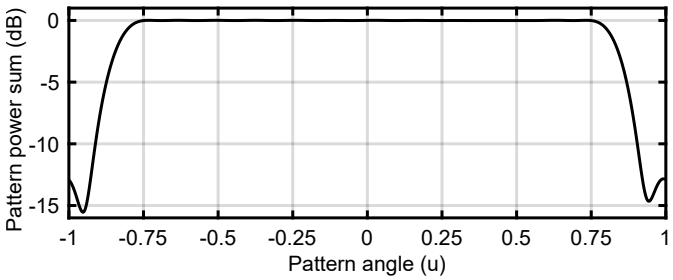


Fig. 5. Sum of the subarray pattern powers for the soft-null compressive array problem in Section IV-C.

independently for both steering and pattern angles, allowing the shape of the sidelobe regions to be controlled.

This problem introduces the idea of a soft stationary null, where the sidelobes are reduced at a fixed pattern angle over all steering angles except those near the soft null itself. This allows a beam to be steered in the direction of the soft null, while suppressing interference from this direction when steering to other angles. This type of array would be useful in scenarios where a high-power source must be suppressed in order to receive weak signals from other directions, but where it is also necessary to monitor the high-power source itself. This situation would occur when low-power radios and a high-power radar must be monitored by a receiver system.

Consider a feed network with $M = 8$ for a ULA with $N = 16$, the start of the sidelobe region at $u_{\mathrm{SLL}} = 0.25$, and a steering range of $|u_s| \leq 1 - u_{\mathrm{SLL}} = 0.75$. This limit prevents any portion of the grating lobes appearing at extreme steering angles in the case of half-wavelength element spacing [1].

A soft null at broadside is required to be 20 dB below the SLL in the remainder of the sidelobe region. The nature of the soft null means that it is only present when $|u_s| \geq 0.25$, so that the null does not enter the predefined main-beam region.

The resulting patterns are shown in Fig. 4. The achieved SLL is -19.85 dB and the 3-dB beamwidth ranges from 0.1837 to 0.2370 depending on the steering angle. The beamwidths narrow near broadside steering due to the nulls at the start of the sidelobe region ($u = \pm 0.25$). The worst pattern level in the soft null region is -39.87 dB, or 20.02 dB below the SLL. Due to the reduced sidelobes at broadside steering, the directivity peaks near broadside steering at 10.14 dB, while the minimum directivity over the steering range is 8.96 dB.

The results were compared to an M-element ULA with the same specifications to ensure that both cases have the same number of beamforming controls. Weights for the ULA were redesigned for each steering angle using the covariance matrix inversion method [1], [14]. The required SLL was specified as -19.85 dB, and the achieved worst-case SLL is -19.76 dB (since no u_{SLL} is specified, the SLL is defined here as the largest pattern value outside the main beams). The largest pattern value in the soft-null region is -39.84 dB below the peak, or 20.08 dB below the achieved SLL. The 3-dB beamwidths range from 0.2465 to 0.2556. Thus, the compressive array improves on the ULA by 7.3% in worst-case beamwidth. The directivity of the ULA ranges from 8.69 dB to 8.85 dB, which is consistently lower than that of the compressive array.

Fig. 5 shows the sum of the M subarray pattern powers over pattern angle, which equals 0 dB $\pm$ 0.01 dB in the steering range as required by (15) and (16). The subarray patterns are suppressed in the out-of-scan regions ($|u| > 0.75$), which explains the higher directivity of the compressive array design over the ULA.

The design of an $M = 8$, $N = 16$ soft-null compressive array, and an 8-element soft-null ULA shows that increasing N for the same M increases the control over the array patterns.

D. Hard Stationary Null Steering

Limiting the steering range of a compressive array means that the subarray patterns may be constrained in any manner outside the steering range. Consider an application where an interfering signal is present from a known direction with significantly higher power than the signal(s) of interest. An example of such a scenario is a passive bistatic radar where the receivers see a direct-path signal from the transmitter of opportunity with a much higher signal strength than the reflected signal from the target, with the direct-path signal as much as 90 dB stronger than the reflected signal [27]. A null formed via digital beamforming (e.g. a soft null) may not suffice, since most of the receivers' available dynamic range will be utilised for sampling the interfering signal to avoid saturation, with little dynamic range left for the signal(s) of interest. A hard stationary null solves this problem by placing a null in the subarray patterns themselves so that the interfering signal is suppressed before sampling.

Consider the constraint function

$$c_{3,m}(\boldsymbol{\phi}_m) = \|\boldsymbol{\phi}_m \mathbf{a}(\theta_w)\|^2 - 10^{W/10} \leq 0 \qquad (20)$$

where $\boldsymbol{\phi}_m$ is the mth row of $\boldsymbol{\Phi}$ such that $b_m(\theta_w) = \boldsymbol{\phi}_m \mathbf{a}(\theta_w)$, θ_w is the desired null direction with $\theta_w \notin [\theta_{s1}, \theta_{s2}]$, and W is the desired magnitude in dB of the mth subarray pattern in the null direction. The null level is not specified relative to the subarray pattern peak, since a subarray pattern does not necessarily have a single distinct peak (Section IV-A). It is specified simply as the value of b_m in the null direction.

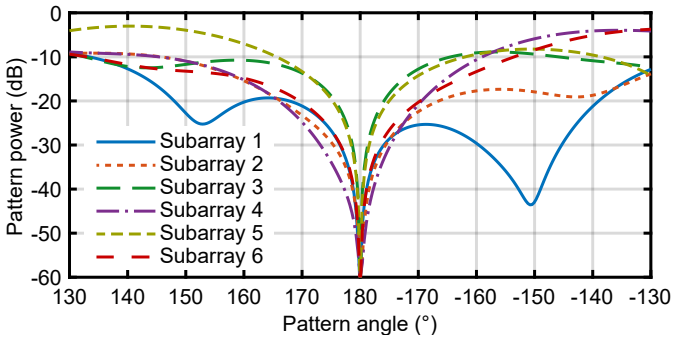


Fig. 6. Superimposed subarray patterns in the out-of-scan region for the compressive circular array problem in Section IV-D.

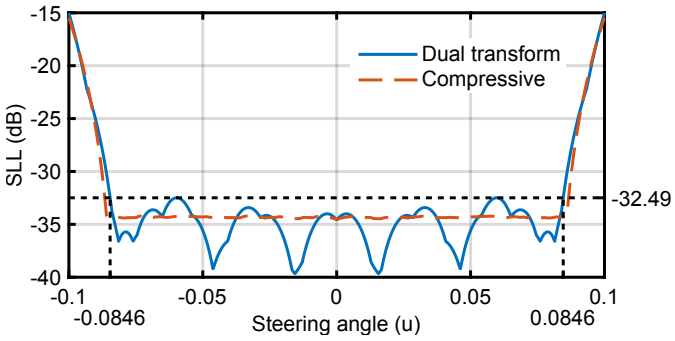


Fig. 7. SLL as a function of steering angle for the dual-transform and compressive arrays in Section IV-E.

Consider the design of a feed network with $M = 6$ for a uniform circular array (UCA) with $N = 9$ elements with half-wavelength spacing between adjacent elements. The steering range is $\theta_s \in [-130°, 130°]$ with a null of magnitude $W = -60$ dB at 180°. The start of the sidelobe region is specified as $\theta_{\text{SLL}} = 50°$.

Fig. 6 shows the resulting subarray patterns, superimposed, in the out-of-scan region with the hard null visible at 180°. The achieved SLL over all steering angles is −13.27 dB and the 3-dB beamwidth ranges from 42.9° to 50.1°.

E. Small Steering Range Array

A dual-transform completely overlapped subarray with $M = 8$ and $N = 64$ is described in [1]. N antenna elements in a ULA are the inputs to an $N \times N$ Butler matrix, but only M outputs corresponding to the beams closest to broadside are used. These M outputs are then passed through an $M \times M$ Butler matrix to produce the M subarray outputs. The subarray outputs are weighted by a 40-dB length-M Chebyshev window. The entire feed network can be described by a fully populated $M \times N$ sensing matrix since each subarray output is a function of the signals at all the antenna elements.

The dual-transform SLLs as a function of steering angle are shown in Fig. 7 (with SLL defined here as the largest pattern value outside the main beam). The specified range of steering angles is $u_s \in [-0.1094, 0.1094]$, but the sidelobes are high at the extremities. As indicated in Fig. 7, the array achieves a worst-case SLL of −32.49 dB over a range of $u_s \in [-0.0846, 0.0846]$ which presents a more useful steering range. The worst 3-dB beamwidth over a steering range of $u_s \in [-0.0846, 0.0846]$ is 0.0398, and the directivity ranges from 16.82 dB to 16.88 dB over the same range.

A compressive array was designed for a slightly wider steering range of $u_s \in [-0.0859, 0.0859]$, with a sidelobe region starting at $u_{\text{SLL}} = 0.0508$ to match the smallest start of a −32.49-dB sidelobe region of the dual-transform array which was found to be 0.0522. The best SLL achieved for the compressive array was −34.20 dB, an improvement of 1.71 dB over the dual-transform array. The worst-case 3-dB beamwidth for the compressive array was 0.0382 which also improves on the dual-transform design. Finally, the directivity varied from 16.97 dB to 17.15 dB which is, again, slightly better than that of the dual-transform array.

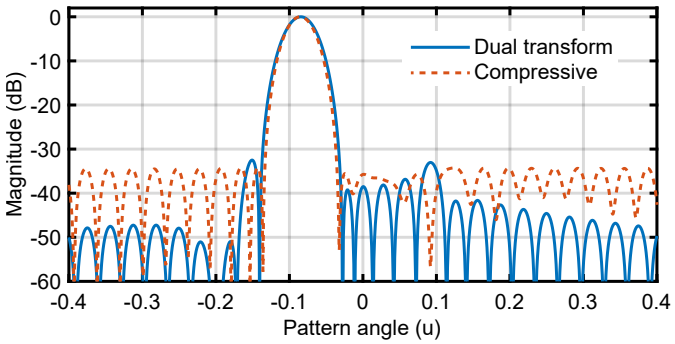


Fig. 8. Compressive array pattern at $u_s = -0.0859$ steering angle and dual-transform array pattern at $u_s = -0.0846$ steering angle for the small steering range problem in Section IV-E.

Fig. 8 shows the patterns for the compressive and dual-transform arrays at steering angles with the worst SLL, namely $u_s = -0.0859$ and $u_s = -0.0846$, respectively. While the dual-transform array has some sidelobes which are lower than those of the compressive array, this characteristic leads to a higher SLL and a wider main beam than the compressive array which has a more uniform pattern in the sidelobe regions.

V. Conclusion

A generalised framework for the optimisation of compressive arrays for low SLL has been proposed. A sequential algorithm for accomplishing this goal, which is based on a coherence-optimisation algorithm, is described.

The proposed algorithm has been validated against a Chebyshev array and achieved the same optimal result. Compressive arrays have been shown to achieve better results than a weighted thinned array designed using the IFT, and improve on the performance of an existing completely overlapped dual-transform subarray. A compressive array with a soft null obtained better results than a conventional ULA with the same number of beamforming controls. The increased number of elements of a compressive array is thus shown to enable better control over the shape of array patterns than a conventional array with the same number of controls. A hard-null compressive array shows that the subarray patterns can be constrained outside the steering range to suppress interference before the controls, for example.

APPENDIX
GOAL AND CONSTRAINT FUNCTION GRADIENTS

Define the function

$$g_{s,a}(\mathbf{\Phi}) = \frac{\left|\mathbf{b}_s^{\mathrm{H}}\mathbf{b}_a\right|^2}{R_{s,a}^2}. \tag{21}$$

It can be shown that the gradients of (21) are given by

$$\frac{\partial g_{s,a}(\mathbf{\Phi})}{\partial \mathbf{\Phi}} = \frac{2}{R_{s,a}^2}\Big[\left(\mathbf{b}_a^{\mathrm{H}}\mathbf{b}_s\right)\mathbf{b}_a\mathbf{a}_s^{\mathrm{H}} + \left(\mathbf{b}_s^{\mathrm{H}}\mathbf{b}_a\right)\mathbf{b}_s\mathbf{a}_a^{\mathrm{H}}\Big]. \tag{22}$$

The gradients of the goal function in (14) are then found as

$$\frac{\partial g(\mathbf{\Phi})}{\partial \mathbf{\Phi}} = \frac{1}{2\alpha C^{1/p}}\left[\sum_{\substack{\theta_a-\theta_s\geq \\ \theta_{\mathrm{SLL}}}}\left(\frac{\left|\mathbf{b}_s^{\mathrm{H}}\mathbf{b}_a\right|}{\alpha R_{s,a}}\right)^p\right]^{1/p-1} \times \sum_{\substack{\theta_a-\theta_s\geq \\ \theta_{\mathrm{SLL}}}}\left[\left(\frac{\left|\mathbf{b}_s^{\mathrm{H}}\mathbf{b}_a\right|}{\alpha R_{s,a}}\right)^{p-2}\frac{\partial g_{s,a}(\mathbf{\Phi})}{\partial \mathbf{\Phi}}\right] \tag{23}$$

with element-wise matrix addition where applicable.

Let $\phi_{m,n}$ represent the elements in $\mathbf{\Phi}$. The gradients of the unit-norm constraint functions (15) to (16) are found using

$$\frac{\partial}{\partial \phi_{m,n}}\left\|\mathbf{b}_s\right\|^2 = 2\mathbf{b}_s\mathbf{a}_s^{\mathrm{H}}. \tag{24}$$

The gradient of the hard-null constraint function (20) is found as

$$\frac{\partial c_{3,m}(\boldsymbol{\phi}_m)}{\partial \phi_{m,n}} = 2b_m(\theta_w)\mathbf{a}(\theta_w)^{\mathrm{H}} \tag{25}$$

for row m in $\mathbf{\Phi}$; the gradients at all other rows are zero.

REFERENCES

[1] R. J. Mailloux, *Phased Array Antenna Handbook*. Norwood, MA, USA: Artech House, 2005.

[2] J. S. Herd and M. D. Conway, "The evolution to modern phased array architectures," *Proc. IEEE*, vol. 104, no. 3, pp. 519–529, Mar. 2016.

[3] C. Fulton, M. Yeary, D. Thompson, J. Lake, and A. Mitchell, "Digital phased arrays: Challenges and opportunities," *Proc. IEEE*, vol. 104, no. 3, pp. 487–503, Mar. 2016.

[4] T. Azar, "Overlapped subarrays: Review and update [Education column]," *IEEE Antennas Propag. Mag.*, vol. 55, no. 2, pp. 228–234, Apr. 2013.

[5] Y. T. Lo, "Aperiodic Arrays," in *Antenna Handbook*, Y. T. Lo and S. W. Lee, Eds. New York, NY: Van Nostrand Reinhold Company Inc., 1988, ch. 14, pp. 14-1–14-37.

[6] W. P. M. N. Keizer, "Linear array thinning using iterative FFT techniques," *IEEE Trans. Antennas Propag.*, vol. 56, no. 8, pp. 2757–2760, Aug. 2008.

[7] M. I. Skolnik, J. W. Sherman, III, and F. C. Ogg, Jr, "Statistically designed density-tapered arrays," *IEEE Trans. Antennas Propag.*, vol. 12, no. 4, pp. 408–417, July 1964.

[8] R. L. Haupt, "Thinned arrays using genetic algorithms," *IEEE Trans. Antennas Propag.*, vol. 42, no. 7, pp. 993–999, July 1994.

[9] W. P. du Plessis, "Weighted thinned linear array design with the iterative FFT technique," *IEEE Trans. Antennas Propag.*, vol. 59, no. 9, pp. 3473–3477, Sep. 2011.

[10] V. Murino, A. Trucco, and A. Tesei, "Beam pattern formulation and analysis for wide-band beamforming systems using sparse arrays," *Signal Process.*, vol. 56, no. 2, pp. 177–183, Jan. 1997.

[11] Y. Wang, G. Leus, and A. Pandharipande, "Direction estimation using compressive sampling array processing," in *IEEE/SP 15th Work. Stat. Signal Process.*, Aug. 2009, pp. 626–629.

[12] H. E. A. Laue and W. P. du Plessis, "Compressive direction-finding antenna array," in *IEEE-APS Topical Conf. on Antennas and Propag. in Wireless Commun. (APWC)*, Sep. 2016, pp. 158–161.

[13] H. E. A. Laue and W. P. du Plessis, "A coherence-based algorithm for optimizing rank-1 Grassmannian codebooks," *IEEE Signal Process. Lett.*, vol. 24, no. 6, pp. 823–827, June 2017.

[14] C. A. Olen and R. T. Compton, "A numerical pattern synthesis algorithm for arrays," *IEEE Trans. Antennas Propag.*, vol. 38, no. 10, pp. 1666–1676, Oct. 1990.

[15] C. L. Dolph, "A current distribution for broadside arrays which optimizes the relationship between beam width and side-lobe level," *Proc. IRE*, vol. 34, no. 6, pp. 335–348, June 1946.

[16] B. Friedlander, "Wireless direction-finding fundamentals," in *Classical and Modern Direction-of-Arrival Estimation*, T. E. Tuncer and B. Friedlander, Eds. Burlingston, MA: Elsevier, 2009, ch. 1, pp. 1–51.

[17] H. E. A. Laue, "Demystifying compressive sensing [Lecture notes]," *IEEE Signal Process. Mag.*, vol. 34, no. 4, pp. 171–176, July 2017.

[18] R. G. Baraniuk, "Compressive sensing [Lecture notes]," *IEEE Signal Process. Mag.*, vol. 24, no. 4, pp. 118–121, July 2007.

[19] S. P. Skobelev, "Methods of constructing optimum phased-array antennas for limited field of view," *IEEE Antennas Propag. Mag.*, vol. 40, no. 2, pp. 39–50, Apr. 1998.

[20] M. Fornasier and H. Rauhut, "Compressive sensing," in *Handbook of Mathematical Methods in Imaging*, O. Scherzer, Ed. New York, NY: Springer New York, 2011, pp. 187–228.

[21] H. Zörlein and M. Bossert, "Coherence optimization and best complex antipodal spherical codes," *IEEE Trans. Signal Process.*, vol. 63, no. 24, pp. 6606–6615, Dec. 2015.

[22] A. Medra and T. N. Davidson, "Flexible codebook design for limited feedback systems via sequential smooth optimization on the Grassmannian manifold," *IEEE Trans. Signal Process.*, vol. 62, no. 5, pp. 1305–1318, Mar. 2014.

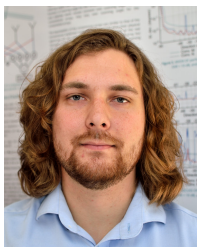
[23] I. S. Dhillon, J. R. Heath, T. Strohmer, and J. A. Tropp, "Constructing packings in Grassmannian manifolds via alternating projection," *Experiment. Math.*, vol. 17, no. 1, pp. 9–35, 2008.

[24] P. Xia, S. Zhou, and G. B. Giannakis, "Achieving the Welch bound with difference sets," *IEEE Trans. Inf. Theory*, vol. 51, no. 5, pp. 1900–1907, May 2005.

[25] J. Nocedal and S. J. Wright, *Numerical optimization*. New York, NY, USA: Springer-Verlag, 1999.

[26] E. J. Candes and M. B. Wakin, "An introduction to compressive sampling," *IEEE Signal Process. Mag.*, vol. 25, no. 2, pp. 21–30, Mar. 2008.

[27] P. E. Howland, D. Maksimiuk, and G. Reitsma, "FM radio based bistatic radar," *IEE Proc.—Radar Sonar Navig.*, vol. 152, no. 3, pp. 107–115, 2005.



Heinrich Laue received the B.Eng. (Electronic) and B.Eng.Hons. (Electronic) degrees, both with distinction, from the University of Pretoria, South Africa in 2015 and 2016, respectively. He is currently pursuing the Ph.D. (Electronic Engineering) degree. His primary research interests are codebook optimisation and the application of CS to antenna arrays.

Warren du Plessis (M'00, SM'10) received the B.Eng. (Electronic) and M.Eng. (Electronic) and Ph.D. (Engineering) degrees from the University of Pretoria in 1998, 2003 and 2010 respectively, winning numerous academic awards including the prestigious Vice-Chancellor and Principal's Medal.

He spent two years as a lecturer at the University of Pretoria, and then joined Grintek Antennas as a design engineer for almost four years, followed by six years at the Council for Scientific and Industrial Research (CSIR). He is currently an Associate Professor at the University of Pretoria, and his primary research interests are cross-eye jamming and thinned antenna arrays.