

Article

# Compressive Sensing Based Design of Sparse Tripole Arrays

Matthew Hawes <sup>1,\*</sup>, Wei Liu <sup>2</sup> and Lyudmila Mihaylova <sup>1</sup>

Received: 17 November 2015; Accepted: 7 December 2015; Published: 10 December 2015

Academic Editor: Vittorio M. N. Passaro

<sup>1</sup> Department of Automatic Control and Systems Engineering, University of Sheffield, Sheffield S1 3JD, UK; l.s.mihaylova@sheffield.ac.uk

<sup>2</sup> Department of Electronic and Electrical Engineering, University of Sheffield, Sheffield S1 3JD, UK; w.liu@sheffield.ac.uk

\* Correspondence: m.hawes@sheffield.ac.uk; Tel./Fax: +44-1142-225-559

**Abstract:** This paper considers the problem of designing sparse linear tripole arrays. In such arrays at each antenna location there are three orthogonal dipoles, allowing full measurement of both the horizontal and vertical components of the received waveform. We formulate this problem from the viewpoint of Compressive Sensing (CS). However, unlike for isotropic array elements (single antenna), we now have three complex valued weight coefficients associated with each potential location (due to the three dipoles), which have to be simultaneously minimised. If this is not done, we may only set the weight coefficients of individual dipoles to be zero valued, rather than complete tripoles, meaning some dipoles may remain at each location. Therefore, the contributions of this paper are to formulate the design of sparse tripole arrays as an optimisation problem, and then we obtain a solution based on the minimisation of a modified  $l_1$  norm or a series of iteratively solved reweighted minimisations, which ensure a truly sparse solution. Design examples are provided to verify the effectiveness of the proposed methods and show that a good approximation of a reference pattern can be achieved using fewer tripoles than a Uniform Linear Array (ULA) of equivalent length.

**Keywords:** sparse array, vector sensors, tripoles, compressive sensing, reweighted minimisation

## 1. Introduction

Sensor arrays are used in a wide range of application areas including Direction Of Arrival (DOA) estimation and beamforming [1], vehicle health maintenance [2] and many others. When considering fixed beamformer design, it is well known that an adjacent antenna separation of less than half a wavelength is required if using Uniform Linear Arrays (ULAs) in order to avoid the introduction of grating lobes [1]. This can become problematic when considering arrays with large apertures due to the cost associated with the number of antennas required. As a result, sparse arrays have become a desirable alternative as they allow a given aperture to be implemented using fewer antennas. The reason for this is that their nonuniform adjacent antenna separations allow mean separations greater than half of the wavelength, while still avoiding the introduction of grating lobes [3].

The tradeoff is that the unpredictable sidelobe behaviour of sparse arrays means that some optimisation of antenna locations is required in order to ensure an acceptable performance is achieved. For example, we could consider minimising the Peak Sidelobe Level (PSL) or matching the response to an acceptable reference response. Stochastic optimisation methods such as Genetic Algorithms (GAs) [4–7] and Simulated Annealing (SA) [8,9] have been shown to be able to solve such a problem. Improved performances can also be achieved by combining GAs with Difference Sets (DS) [10] and Almost Difference Sets (ADS) [11]. However, the disadvantages of GAs and similar methods are the potentially long computation times and the possibility of failing to converge to the optimal solution.

Instead, Compressive Sensing (CS) [12] (or alternatively the Bayesian formulation [13]) tells us that, when certain conditions are met, we can recover signals from fewer measurements than are traditionally required. CS-based methods have been applied to various array signal processing problems such as DOA estimation [14] and in the design of sparse antenna arrays [15–19]. For CS-based sparse array design methods, the aim is to use as few antennas as possible, while still getting an exact, or almost exact, match to a reference response. It has further been shown that the sparsity of the solution can be improved by converting the problem into a series of iteratively solved reweighted minimisations [20,21]. This is achieved by making the problem a closer approximation of an  $l_0$  norm minimisation by adding a reweighting term that penalises small weight coefficients more heavily [22].

Previous work involving CS and sparse array design has tended to focus on the case of isotropic array elements. This means that the polarisation of the signal of interest is not accounted for, potentially degrading the performance of an array. Alternatively, we can consider arrays consisting of vector sensors allowing measurement of different components of the received electromagnetic waveform [23–26]. One such vector sensor is a tripole, which consists of three orthogonally orientated dipoles [27–35]. Such vector sensors have been used in various configurations for DOA and polarisation estimation [36–43]. However, in these works, fixed sensor locations are considered rather than addressing the location optimisation problem associated with sparse array design.

The novelty of this paper arises in the proposed solution to the problem of designing sparse tripole arrays using CS for the first time. Unlike for isotropic array elements (consisting of array elements of a single antenna), we now have three complex valued weight coefficients (one for each co-located dipole) that have to be simultaneously minimised for each potential tripole location. If this is not done, there is the potential for only the coefficients of single dipoles (rather than complete tripoles) to be made zero-valued. As a result, we can obtain a solution with at least one dipole at each location on the grid of potential sensor locations rather than obtaining a truly sparse solution. Instead, to ensure the desired sparsity is achieved, we propose two CS-based methods of solving this problem: (i) the optimisation can be formulated as a modified  $l_1$  norm minimisation; (ii) the sparsity can be further improved by converting the problem into a series of iteratively solved reweighted minimisations. We provide broadside and off-broadside design examples to show that the proposed methods can successfully obtain a good approximation of a reference pattern using fewer tripoles than a ULA of equivalent length.

This work differs from previous work of the first and third authors as follows: (i) In [44–46], we considered the problem of designing sparse sensor arrays for both narrowband and wideband beamforming. Unlike the current work, the arrays consisted of isotropic array elements with a single sensor at each location. In [47], we dealt with the design of sparse linear arrays based on crossed dipoles. However, [47] deals with a quaternionic signal model, which is not utilised in this work; (ii) [48] deals with DOA estimation using a fixed spatially stretched tripole (no co-located tripole) rather than the sparse array design problem.

The remainder of this paper is structured as follows, Section 2 gives details of the proposed design methods including: the array model in Section 2.1 and the CS based formulations in Section 2.2. Design examples are presented in Section 3, with conclusions being drawn in Section 4.

## 2. Proposed Design Methods

### 2.1. Array Model

Figure 1 shows the tripole array structure being considered. There are  $M$  tripole locations spread over the  $y$ -axis with an adjacent separation distance  $d$ . For each tripole, there are three orthogonally orientated dipoles, one parallel to each axis. Also shown is a signal with its DOA defined by the angles  $\theta$  and  $\phi$ , which are limited as follows:  $0 \leq \theta \leq \pi/2$  and  $-\pi/2 \leq \phi \leq \pi/2$ . A plane-wave signal model is assumed, *i.e.*, the signal impinges upon the array from the far field.

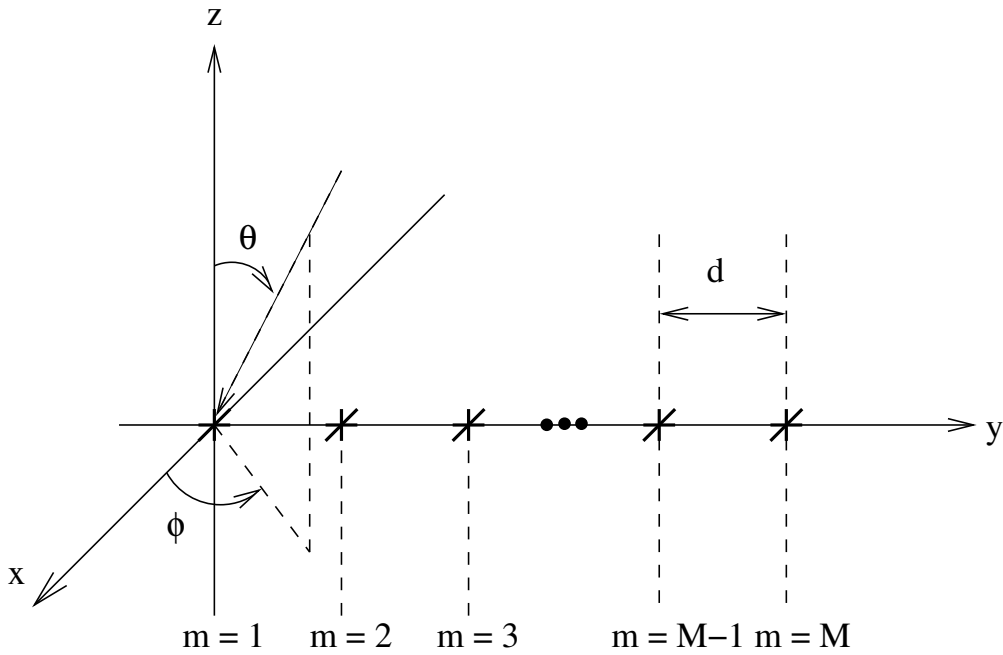


Figure 1. An array model with  $M$  potential tripole locations shown.

The spatial steering vector of the array is given by

$$\mathbf{s}_s(\theta, \phi) = [1, e^{-j2\pi d \sin \theta \sin \phi / \lambda}, \dots, e^{-j2\pi(M-1)d \sin \theta \sin \phi / \lambda}] \quad (1)$$

where  $\lambda$  is the wavelength of the signal of interest. For tripoles, the spatial-polarization coherent vector contains information about a signal's polarisation and is given by [27]:

$$\begin{aligned} \mathbf{s}_p(\theta, \phi, \gamma, \eta) &= \begin{bmatrix} \sin \gamma \cos \theta \cos \phi e^{j\eta} - \cos \gamma \sin \phi \\ \sin \gamma \cos \theta \sin \phi e^{j\eta} - \cos \gamma \cos \phi \\ -\sin \gamma \sin \theta e^{j\eta} \end{bmatrix} \\ &= \begin{bmatrix} s_{p,x}(\theta, \phi, \gamma, \eta) \\ s_{p,y}(\theta, \phi, \gamma, \eta) \\ s_{p,z}(\theta, \phi, \gamma, \eta) \end{bmatrix} \end{aligned} \quad (2)$$

where  $0 \leq \gamma \leq \pi/2$  is the auxiliary polarization angle and  $-\pi \leq \eta < \pi$  is the polarization phase difference.

Now, the array can be split into three sub-arrays, one parallel to each axis. The steering vector of each of these sub-arrays is complex-valued and given by:

$$\mathbf{s}_x(\theta, \phi, \gamma, \eta) = s_{p,x}(\theta, \phi, \gamma, \eta) \mathbf{s}_s(\theta, \phi) \quad (3)$$

$$\mathbf{s}_y(\theta, \phi, \gamma, \eta) = s_{p,y}(\theta, \phi, \gamma, \eta) \mathbf{s}_s(\theta, \phi) \quad (4)$$

and

$$\mathbf{s}_z(\theta, \phi, \gamma, \eta) = s_{p,z}(\theta, \phi, \gamma, \eta) \mathbf{s}_s(\theta, \phi) \quad (5)$$

respectively.

The response of the array is given by

$$p(\theta, \phi, \gamma, \eta) = \mathbf{s}(\theta, \phi, \gamma, \eta) \mathbf{w}^H \quad (6)$$

with

$$\mathbf{w} = [w_{x,1}, w_{y,1}, w_{z,1}, \dots, w_{x,M}, w_{y,M}, w_{z,M}] \quad (7)$$

where  $w_{i,m}$  is the weight coefficient for the  $m^{\text{th}}$  dipole orientated parallel to the  $i$ -axis ( $i = \{x, y, z\}$ ) and  $\{\cdot\}^H$  denotes the Hermitian transpose. Similarly, we also have

$$\begin{aligned} \mathbf{s}(\theta, \phi, \gamma, \eta) = & [s_{x,1}(\theta, \phi, \gamma, \eta), s_{y,1}(\theta, \phi, \gamma, \eta), \\ & s_{z,1}(\theta, \phi, \gamma, \eta), \dots, s_{x,M}(\theta, \phi, \gamma, \eta), \\ & s_{y,M}(\theta, \phi, \gamma, \eta), s_{z,M}(\theta, \phi, \gamma, \eta)] \end{aligned} \quad (8)$$

where  $s_{i,m}(\theta, \phi, \gamma, \eta)$  is the contribution of the  $m^{\text{th}}$  dipole to the overall steering vector parallel to the  $i$ -axis.

In what follows, we address the problem of designing a sparse linear tripole array, where there are two issues to be solved. Firstly, we have to find the optimal tripole locations, as there will no longer be a uniform adjacent tripole separation of  $d$ . Secondly, we have to find the weight coefficients that give an acceptable beam response. We propose to use design methods based on CS to solve these problems.

## 2.2. Compressive Sensing Based Design of Sparse Tripole Arrays

Suppose  $P_r(\theta, \phi, \gamma, \eta)$  is a reference response which we wish to achieve. First, consider Figure 1 as being a grid of potential tripole locations. In this instance,  $(M - 1)d$  is the potential aperture of the array and  $M$  is a large number. Sparseness is then introduced by selecting the weight coefficients to give as few active tripoles as possible, or, in other words, as few non-zero valued weight coefficients as possible. This has to be done while still giving a designed response that is close to the desired one.

This problem is formulated as

$$\begin{aligned} \min \quad & \|\mathbf{w}\|_1 \\ \text{subject to} \quad & \|\mathbf{p}_r - \mathbf{S}\mathbf{w}^H\|_2 \leq \alpha \end{aligned} \quad (9)$$

where  $\|\mathbf{w}\|_1$  is the  $l_1$  norm of the weight coefficients and used as an estimate of the number of nonzero weight coefficients in  $\mathbf{w}$ , *i.e.*, an approximation to the  $l_0$  norm,  $\mathbf{p}_r$  is the vector holding the desired beam response at the sampled angular and polarisation points of interest,  $\mathbf{S}$  is the matrix composed of the corresponding steering vectors and  $\alpha$  places a limit on the allowed difference between the desired and the designed responses. Therefore, by minimising  $\|\mathbf{w}\|_1$  in Equation (9), we are minimising the number of tripoles we have to use, while the constraint ensures that we still get a good approximation of the reference response.

In detail,  $\mathbf{p}_r$  and  $\mathbf{S}$  are respectively given by

$$\mathbf{p}_r = [P_r(\theta_1, \phi_1, \gamma_1, \eta_1), \dots, P_r(\theta_L, \phi_L, \gamma_L, \eta_L)]^T \quad (10)$$

$$\mathbf{S} = [\mathbf{s}(\theta_1, \phi_1, \gamma_1, \eta_1), \dots, \mathbf{s}(\theta_L, \phi_L, \gamma_L, \eta_L)]^T \quad (11)$$

where  $L$  is the number of points sampled at each dimension of the desired beam response and  $\{\cdot\}^T$  denotes the transpose operation. In this work,  $\mathbf{p}_r$  is the ideal response, *i.e.*, a value of one for the mainlobe and zeros for the other entries. It would also be possible to use other reference responses if desired, for example, a response which defines a maximum sidelobe level or one with nulls at a given direction.

Note, we have to choose  $L$  to be large enough to ensure that all angular and polarisation points of interest are being considered. However, the larger  $L$  is, the longer the computation time will be. An acceptable compromise seems to be to sample the sidelobe regions every  $1^\circ$ . We can also help improve the sparsity of the solution by increasing the value of  $M$ . A larger value of  $M$  gives a denser grid which is more likely to include the optimal locations. As a result, the same amount of error between designed and reference responses can be obtained using fewer tripoles. However, we will get to the

point where the optimal locations will be already included on the grid and further increases will only serve to lengthen the computation time. This means there is again a compromise to be achieved when selecting the value of  $M$ , and this will be explored in the design examples we present in this paper.

One problem with the above formulation is that the three complex weight coefficients associated with each tripole can not be guaranteed to be minimised simultaneously. As a result, we may not be able to obtain a sparse solution even if we have a sparse weight vector  $\mathbf{w}$  after the minimisation. In other words, it would be possible to end up only removing a single dipole from each potential location rather than truly introducing sparsity by removing complete tripoles. Therefore, we further modify the formulation in Equation (9) by converting it into a modified  $l_1$  norm minimisation.

First, we rewrite Equation (9) as

$$\begin{aligned} \min \quad & t \in R^+ \\ \text{subject to} \quad & \|\mathbf{p}_r - \mathbf{S}\mathbf{w}^H\|_2 \leq \alpha, \quad |\langle \mathbf{w} \rangle|_1 \leq t \end{aligned} \tag{12}$$

where

$$|\langle \mathbf{w} \rangle|_1 = \sum_{m=1}^M \|\mathbf{w}_m\|_2 \tag{13}$$

and  $\mathbf{w}_m = [w_{x,m}, w_{y,m}, w_{z,m}]$ . Here  $\mathbf{w}_m$  contains the weight coefficients for all the dipoles that make up the tripole at the  $m^{\text{th}}$  grid location. Therefore, by minimising the value of  $t$  the second constraint in Equation (12) ensures the combined weight coefficients for the tripoles are also minimised.

Now, we decompose  $t$  to  $t = \sum_{m=1}^M t_m, t_m \in R^+$ . This allows us to minimise the combined weight coefficients for each tripole separately. In vector form, we have

$$t = [1, \dots, 1] \begin{bmatrix} t_1 \\ \vdots \\ t_M \end{bmatrix} = \mathbf{1}^T \mathbf{t}. \tag{14}$$

Then, Equation (12) can be rewritten as

$$\begin{aligned} \min_{\mathbf{t}} \quad & \mathbf{1}^T \mathbf{t} \\ \text{subject to} \quad & \|\mathbf{p}_r - \mathbf{S}\mathbf{w}^H\|_2 \leq \alpha \\ & \|\mathbf{w}_m\|_2 \leq t_m, \quad m = 1, \dots, M. \end{aligned} \tag{15}$$

Now, define

$$\hat{\mathbf{w}} = [t_1, R(w_{x,1}), R(w_{y,1}), R(w_{z,1}), -I(w_{x,1}), -I(w_{y,1}), -I(w_{z,1}), t_2, \dots, -I(w_{z,M})], \tag{16}$$

$$\hat{\mathbf{c}} = [1, 0, 0, 0, 0, 0, 0, 1, 0, 0, \dots, 1, 0, 0, 0, 0, 0]^T, \tag{17}$$

$$\hat{\mathbf{p}}_r = [\mathbf{R}(\mathbf{p}_r), \mathbf{I}(\mathbf{p}_r)]^T \tag{18}$$

and

$$\hat{\mathbf{S}} = \begin{pmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{R}(\mathbf{s}_{x,1}) & \mathbf{I}(\mathbf{s}_{x,1}) \\ \mathbf{R}(\mathbf{s}_{y,1}) & \mathbf{I}(\mathbf{s}_{y,1}) \\ \mathbf{R}(\mathbf{s}_{z,1}) & \mathbf{I}(\mathbf{s}_{z,1}) \\ -\mathbf{I}(\mathbf{s}_{x,1}) & \mathbf{R}(\mathbf{s}_{x,1}) \\ -\mathbf{I}(\mathbf{s}_{y,1}) & \mathbf{R}(\mathbf{s}_{y,1}) \\ -\mathbf{I}(\mathbf{s}_{z,1}) & \mathbf{R}(\mathbf{s}_{z,1}) \\ \vdots & \vdots \\ \mathbf{0} & \mathbf{0} \\ \vdots & \vdots \\ -\mathbf{I}(\mathbf{s}_{z,M}) & \mathbf{R}(\mathbf{s}_{z,M}) \end{pmatrix}^T \tag{19}$$

where  $\mathbf{s}_{i,m}$  is the contribution of the  $m^{th}$  dipole parallel to the  $i$ -axis to the steering vector of the array,  $R(\cdot)$  gives the real component of a scalar and  $I(\cdot)$  gives the imaginary component of a scalar and  $\mathbf{R}(\cdot)$  gives the real components of a vector/matrix and  $\mathbf{I}(\cdot)$  gives the imaginary components of a vector/matrix. Here, we have included the values of  $t_m$  in  $\hat{\mathbf{w}}$  as we want to automatically find them with the weight coefficients instead of them being predetermined. We then use  $\hat{\mathbf{c}}$  to select the values of  $t_m$  for minimisation, and the zeros are added in  $\hat{\mathbf{S}}$  to ensure they do not contribute to the error between the reference and designed responses.

Then, the final formulation is as follows

$$\begin{aligned} & \min_{\hat{\mathbf{w}}} \quad \hat{\mathbf{w}}\hat{\mathbf{c}} \\ \text{subject to} \quad & \|\hat{\mathbf{p}}_r - \hat{\mathbf{S}}\hat{\mathbf{w}}^T\|_2 \leq \alpha \\ & \|\mathbf{w}_m\|_2 \leq t_m, \quad m = 1, \dots, M \end{aligned} \tag{20}$$

which can be solved using *cvx*, a package for specifying and solving convex programs [49,50].

As with the traditional CS based design methods, the solution can be improved by considering the problem as a series of iteratively solved reweighted minimisations to bring the solution closer to that of the  $l_0$  norm minimisation [20–22]. This is achieved by the addition of a reweighting term which is found from the previous iteration and penalises small non-zero valued weight coefficients more heavily, meaning all non-zero valued coefficients are treated in a more uniform manner (as for the  $l_0$  norm minimisation). As a result, these small non-zero valued weight coefficients are less likely to be repeated in the next iteration and the sparsity of the solution is improved.

In this form, the problem is formulated as follows

$$\begin{aligned} & \min_{\hat{\mathbf{w}}} \quad \hat{\mathbf{w}}\hat{\mathbf{c}} \\ \text{subject to} \quad & \|\hat{\mathbf{p}}_r - \hat{\mathbf{S}}\hat{\mathbf{w}}^T\|_2 \leq \alpha \\ & \delta_m^k \|\mathbf{w}_m\|_2 \leq t_m, \quad m = 1, \dots, M \end{aligned} \tag{21}$$

where we now have

$$\hat{\mathbf{c}} = [\delta_1^k, 0, 0, 0, 0, 0, 0, \delta_2^k, 0, \dots, \delta_M^k, 0, 0, 0, 0, 0]^T \tag{22}$$

and

$$\delta_m^k = (|\mathbf{w}_m^{k-1}| + \epsilon)^{-1}. \tag{23}$$

This is then solved iteratively until the number of non-zero valued weight coefficients has remained constant for a few iterations.

Note, the term  $\epsilon$  is added for numerical stability and should be set slightly smaller than the minimum implemented weight coefficient for a single tripole. However, its addition means that a

zero-valued weight coefficient in one iteration will not be guaranteed to be repeated in the next. This has the result of meaning a solution is not always guaranteed, but when a solution is possible, it will usually be achieved in less than 10 iterations. For the first iteration, Equation (20) can be solved and the reweighting terms introduced from the second iteration onwards. Finally,  $\mathbf{w}_m$  gives the current estimate of the weight coefficients for the  $m^{\text{th}}$  tripole, whereas  $\mathbf{w}_m^{k-1}$  gives the estimate from the previous iteration.

### 3. Design Examples

We will now consider broadside and off-broadside design examples to verify the effectiveness of the two design methods. All examples are implemented on a computer with an Intel Xeon CPU E3-1271 (3.60 GHz), 16GB of RAM and Windows 7 operating system, using Matlab R2014a and cvx [49,50]. In all of the figures that follow positive values of  $\theta$  indicate the value range  $\theta \in [0^\circ, 90^\circ]$  with  $\phi = 90^\circ$ , while negative values of  $\theta \in [-90^\circ, 0^\circ]$  indicate an equivalent range of  $\theta \in [0^\circ, 90^\circ]$  with  $\phi = -90^\circ$ .

#### 3.1. Broadside Design Examples

First, we will consider broadside ( $\theta_{ML} = 0^\circ, \phi_{ML} = 90^\circ$ ) design examples for both methods and show how the value of  $M$  effects their performances. The length of the array is set to be 10 $\lambda$  and the values  $\alpha = 0.5, \gamma = 55^\circ$  and  $\eta = 100^\circ$  selected. Finally, the sidelobe regions are defined as  $\theta_{SL} = [10^\circ, 90^\circ]$  for  $\phi_{SL} = \pm 90^\circ$ .

Table 1 compares the performances of the proposed methods for values of  $M = 101, 301$  and 501. For each case, the results for the non-reweighted example are found using Equation (20) and for the reweighted example Equation (21) is used. Here, we are comparing the aperture of the designed array, number of tripoles required, mean adjacent tripole separation ( $\overline{\Delta d}$ ), the minimum adjacent tripole separation, the computation time, designed mainlobe location ( $\hat{\theta}_{ML}$ ) and PSL. The first three give an indication of the level of sparsity introduced by the design methods, the computation time is used to indicate the implementation complexity and the final two measures give an indication of how desirable the final response is. Note, we are not using  $\|\mathbf{p}_r - \mathbf{S}\mathbf{w}^H\|_2$  as a performance measure as the same value of  $\alpha$  is used in each case, meaning the value of  $\|\mathbf{p}_r - \mathbf{S}\mathbf{w}^H\|_2$  will be the same for both methods.

**Table 1.** Broadside performance comparison.

Example	Aperture/ $\lambda$	Number of Tripoles	$\overline{\Delta d}/\lambda$	Min( $\Delta d$ )	Computation Time (s)	$\hat{\theta}_{ML}$	PSL (dB)
Non-reweighted M= 101	10	16	0.67	0.15	5.61	90°	−32.28
Reweighted M= 101	5.20	8	0.74	0.60	25.92	90°	−23.25
Non-reweighted M= 301	10	14	0.77	0.57	23.86	90°	−32.53
Reweighted M= 301	5.13	8	0.73	0.60	88.88	90°	−22.94
Non-reweighted M= 501	10	14	0.77	0.58	49.90	90°	−32.69
Reweighted M= 501	5.12	8	0.73	0.60	168.52	90°	−22.89

The first thing to note is that, for all values of  $M$ , both methods have successfully obtained a good approximation of the reference pattern and achieved a desirable beam response. This is evident from the fact that the mainlobes are always in the correct location with sufficient sidelobe attenuation simultaneously achieved.

When looking at the results for the non-reweighted examples, we can see that it is possible, but not guaranteed, that increasing the value of  $M$  can help increase the sparsity. This can be explained by considering that the denser the grid of potential tripole locations, the more likely it is that the grid



will include the optimal tripole locations. As a result, it would be possible to achieve the same error between reference and designed responses using fewer tripoles. However, we will get to the point where the grid is dense enough to include the optimal locations and further increasing the value of  $M$  has little or no effect. This is evident from the observation that increasing  $M$  from 101 to 301 has increased the mean and minimum adjacent separations as well as decreasing the number of tripoles required. In other words, the sparsity has been increased. However, when increasing  $M$  further to 501 only the minimum adjacent sensor separation has increased, showing less of an improvement has been achieved. It is also worth noting that increasing  $M$  has been shown to always increase the computation time. If we take the computation time for  $M = 101$  as a reference, we can see that there is a 425% increase when  $M = 301$  and a 978% increase in computation time when  $M = 501$ . Similarly, for the reweighted examples, we have increases of 343% for  $M = 301$  and 650% for  $M = 501$ . As a result, a balance has to be achieved rather than continually increasing  $M$ . In this case,  $M = 301$  seems to be a good choice.

From the results for  $M = 101$ , we can see that using the reweighted method has improved all measures of sparsity of the result (*i.e.*, number of tripoles, mean and minimum adjacent tripole separation). For  $M = 301$  and  $M = 501$  we can see that the reweighted method still gives a significant improvement in the number of tripoles that are required. As a result, a greater cost saving is made. However, the iterative nature of this method means that in each case there has been an increase in the computation time, as would be expected. For  $M = 101$  we have a 462% increase, 373% for  $M = 301$  and 338% for  $M = 501$ .

As an example, the tripole locations for each method when  $M = 301$  are shown in Tables 2 and 3, respectively, with Figure 2 showing the designed responses. For completeness a ULA with adjacent tripole separation of  $\lambda/2$  and aperture of  $10\lambda$  is also shown for a comparison. The weight coefficients for this comparison ULA can be found using the following:

$$\begin{aligned} \min \quad & \|\hat{\mathbf{p}}_r - \hat{\mathbf{S}}_{ULA} \hat{\mathbf{w}}_{ULA}^T\|_2 \\ \text{subject to} \quad & \hat{\mathbf{S}}_{ULA, \theta_{ML}} \hat{\mathbf{w}}_{ULA}^T = 1 \end{aligned} \tag{24}$$

where  $\mathbf{S}_{ULS}$  is given by

$$\hat{\mathbf{S}}_{ULA} = \begin{pmatrix} \mathbf{R}(\mathbf{s}_{ULA,x,1}) & \mathbf{I}(\mathbf{s}_{ULA,x,1}) \\ \mathbf{R}(\mathbf{s}_{ULA,y,1}) & \mathbf{I}(\mathbf{s}_{ULA,y,1}) \\ \mathbf{R}(\mathbf{s}_{ULA,z,1}) & \mathbf{I}(\mathbf{s}_{ULA,z,1}) \\ -\mathbf{I}(\mathbf{s}_{ULA,x,1}) & \mathbf{R}(\mathbf{s}_{ULA,x,1}) \\ -\mathbf{I}(\mathbf{s}_{ULA,y,1}) & \mathbf{R}(\mathbf{s}_{ULA,y,1}) \\ -\mathbf{I}(\mathbf{s}_{ULA,z,1}) & \mathbf{R}(\mathbf{s}_{ULA,z,1}) \\ \vdots & \vdots \\ -\mathbf{I}(\mathbf{s}_{ULA,z,M}) & \mathbf{R}(\mathbf{s}_{ULA,z,M}) \end{pmatrix}^T \tag{25}$$

$\mathbf{s}_{ULA,i,m}$  is the contribution of the  $m^{th}$  dipole parallel to the  $i$ -axis,  $\hat{\mathbf{S}}_{ULA, \theta_{ML}}$  is  $\hat{\mathbf{S}}_{ULA}$  just for the mainlobe,

$$\begin{aligned} \hat{\mathbf{w}}_{ULA} = & [R(w_{ULA,x,1}), R(w_{ULA,y,1}), R(w_{ULA,z,1}), -I(w_{ULA,x,1}), -I(w_{ULA,y,1}), \\ & -I(w_{ULA,z,1}), \dots, -I(w_{ULA,z,M})] \end{aligned} \tag{26}$$

and  $w_{ULA,i,m}$  is the weight coefficient for the  $m^{th}$  dipole parallel to the  $i$ -axis. We can view this as finding the weight coefficients  $\hat{\mathbf{w}}_{ULA}$  that minimise the difference between the reference and designed responses, subject to keeping a unity response for the mainlobe direction. In practice, this has the effect of ensuring a distortionless response at the mainlobe while minimising the sidelobes.

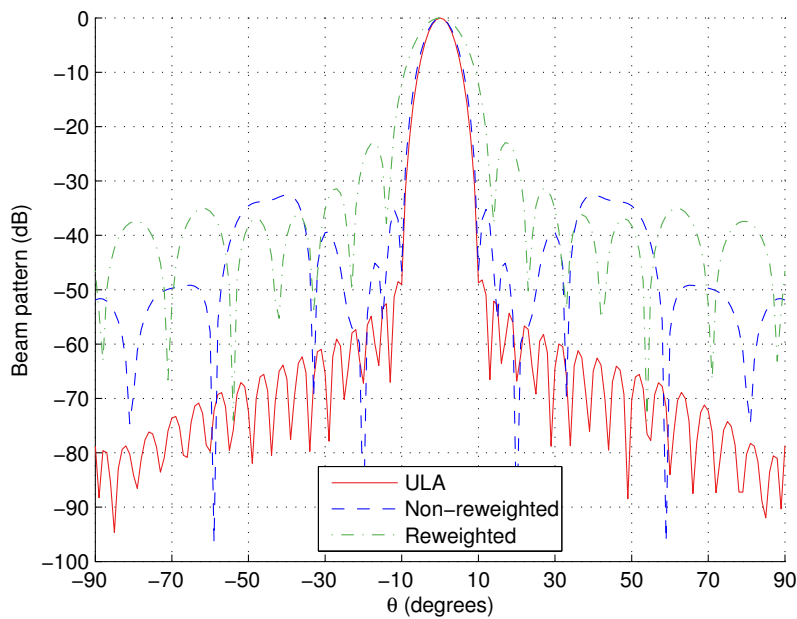


**Table 2.** Locations for the non-reweighted broadside design example,  $M = 301$ .

$n$	1	2	3	4	5	6	7	8	9	10	11	12	13	14
$d_n/\lambda$	0	0.80	1.62	2.42	3.25	4.02	4.72	5.28	5.98	6.75	7.58	8.38	9.20	10

**Table 3.** Locations for the reweighted broadside design example,  $M = 301$ .

$n$	1	2	3	4	5	6	7	8
$d_n/\lambda$	2.43	3.23	4.03	4.70	5.30	5.97	6.77	7.57



**Figure 2.** Responses for broadside design example,  $M = 301$ .

Note, Figure 2 shows that the responses for the arrays designed using the two proposed methods are slightly less desirable than the comparison ULA. This is to be expected, as significantly less tripoles are being used (21 for the ULA compared to 14 or 8 for the proposed methods). However, they still give acceptable approximations of the reference response. Here, the important thing is the reduction in the number of tripoles as was the aim of the proposed methods.

### 3.2. Off-Broadside Design Examples

We will now present off-broadside design examples for both of the proposed methods. Here, we will use the parameters from the broadside design example but with the value of  $M = 301$  fixed. The performance will be considered over a range of desired mainlobe locations ( $\theta_{ML}$  varies for a fixed  $\phi_{ML} = 90^\circ$ ) and, in each case, there will be a transition region of  $10^\circ$  on either side of the mainlobe with the remaining regions being the sidelobes.

Table 4 summarises the performance of the proposed methods over the range of mainlobes being considered. We can see that an acceptable approximation of the reference response is achieved in all cases (*i.e.*, mainlobe is within  $1^\circ$  of what is designed and sufficient sidelobe attenuation is achieved). In addition, for each example, using the reweighted method has reduced the number of tripoles required, but at the expense of an increased computation time (321% for  $\theta = 10^\circ$ , 295% for  $\theta = 30^\circ$ , 582% for  $\theta = 50^\circ$  and 544% for  $\theta = 70^\circ$ ). In the case of the  $\theta_{ML} = 50^\circ$  and  $\theta_{ML} = 70^\circ$  examples, this reduction in tripoles has even made the difference between whether there are any savings in number of tripoles as compared to an equivalent length ULA or not. However, for the non-reweighted method, we can

also get a result with improved sparsity by increasing the value of  $\alpha$ , although this will mean that the designed response will be a worse approximation of the reference response as a result. For example, see Table 5, where a value of  $\alpha = 0.95$  is shown to give a reduction in the number tripoles required when using the non-reweighted method.

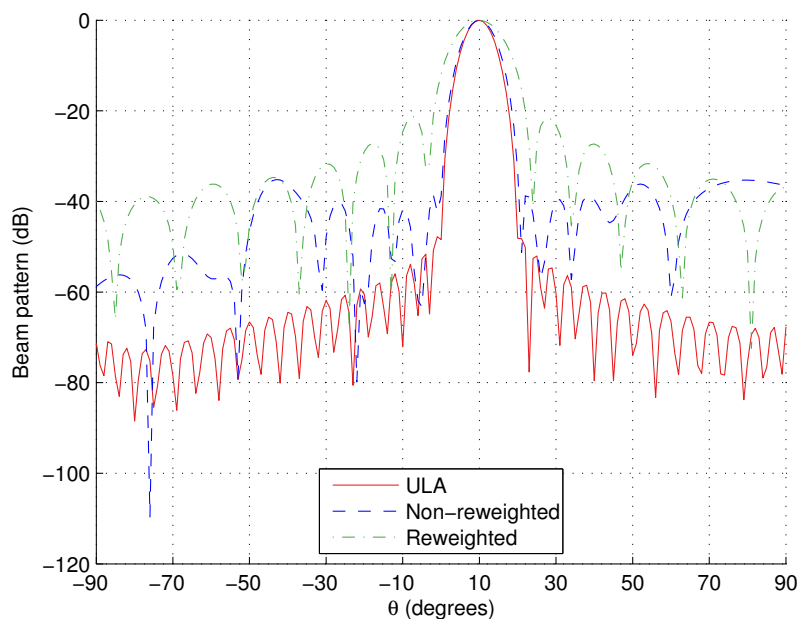
**Table 4.** Off-broadside performance comparison, for  $\alpha = 0.5$ .

$\theta_{ML}$	Aperture/ $\lambda$	Number of Tripoles	$\overline{\Delta d}/\lambda$	Min( $\Delta d$ )	Computation Time (s)	$\hat{\theta}_{ML}$	PSL (dB)
Non-reweighted 10°	10	15	0.71	0.68	22.17	10°	−35.18
Reweighted 10°	5.73	9	0.72	0.70	71.17	10°	−21.27
Non-reweighted 30°	10	18	0.59	0.55	23.40	30°	−30.92
Reweighted 30°	6.53	12	0.59	0.57	69.05	29°	−23.27
Non-reweighted 50°	10	23	0.45	0.35	24.88	50°	−24.69
Reweighted 50°	8.20	6	1.64	1.57	144.89	49°	−18.12
Non-reweighted 70°	10	24	0.43	0.17	21.18	69°	−19.61
Reweighted 70°	10	8	1.43	0.17	115.22	70°	−17.51

**Table 5.** Off-broadside performance comparison, for  $\alpha = 0.95$ .

$\theta_{ML}$	Aperture/ $\lambda$	Number of Tripoles	$\overline{\Delta d}/\lambda$	Min( $\Delta d$ )	Computation Time (s)	$\hat{\theta}_{ML}$	PSL (dB)
Non-reweighted 50°	10	18	0.59	0.48	25.25	50°	−15.23
Non-reweighted 70°	10	20	0.53	0.33	25.44	70°	−7.59

Here, we show the responses for  $\theta_{ML} = 10^\circ$  as an example in Figure 3. Again, the response for a ULA of aperture  $10\lambda$  and adjacent tripole separation of  $\lambda/2$  is shown for a comparison. As for the broadside example with  $M = 301$ , we can see that the designed responses are again slightly less desirable (wider mainlobe and increased PSL). However, this is an acceptable cost for achieving the desired reduction in the number of tripoles used. For completeness, the resulting tripole locations are shown in Tables 6 and 7, respectively.



**Figure 3.** Responses for off-broadside design example,  $\theta_{ML} = 10^\circ$ .

**Table 6.** Locations for the non-reweighted off-broadside design example,  $\theta_M = 10^\circ$ .

<b>n</b>	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
$d_n/\lambda$	0	0.68	1.40	2.12	2.85	3.55	4.28	5	5.72	6.45	7.15	7.88	8.60	9.32	10

**Table 7.** Locations for the reweighted off-broadside design example,  $\theta_M = 10^\circ$ .

<b>n</b>	1	2	3	4	5	6	7	8	9
$d_n/\lambda$	2.13	2.83	3.57	4.27	5	5.73	6.43	7.17	7.87

#### 4. Conclusions

This paper has addressed the problem of designing sparse linear tripole arrays for the first time. Unlike for isotropic array elements, each antenna now has three complex valued weight coefficients (one for each co-located dipole) which have to be simultaneously minimised in order to guarantee a truly sparse solution. Therefore, we have proposed two CS-based solutions to this problem: (i) a modified  $l_1$  norm minimisation; (ii) a series of iteratively solved reweighted minimisations, which give an improved level of sparsity. Both broadside and off-broadside design examples have been provided to verify the effectiveness of the proposed methods. In both cases, the results show that a good approximation of the reference pattern can be achieved using fewer tripoles than a ULA of equivalent length. Note, in this paper, we have focused on the case of a single polarisation of interest, as we were more interested in showing it was possible to design sparse tripole arrays using CS rather than dealing with multiple polarisations. In order to handle multiple polarisations, it may be necessary to extend the proposed methods to designing planar arrays, which is seen as an area for future research.

**Acknowledgments:** We appreciate the support of the UK Engineering and Physical Sciences Research Council (EPSRC) via the project Bayesian Tracking and Reasoning over Time (BTaRoT) grant EP/K021516/1.

**Author Contributions:** Matthew Hawes developed the proposed design methods and wrote the manuscript. Wei Liu formulated the original problem being considered. Wei Liu and Lyudmila Mihaylova were involved in the theoretical development of the solutions, in editing and structuring the manuscript. All authors were involved in discussing potential solutions, the results and determining the best test scenarios to be used.

**Conflicts of Interest:** The authors declare no conflict of interest.

#### References

1. Van Trees, H.L. *Optimum Array Processing, Part IV of Detection, Estimation, and Modulation Theory*; John Wiley & Sons, Inc.: New York, NY, USA, 2002.
2. Rodger, J.A. Toward reducing failure risk in an integrated vehicle health maintenance system: A fuzzy multi-sensor data fusion kalman filter approach for IVHMS. *Expert Syst. Appl.* **2012**, *39*, 9821–9836.
3. Jarske, P.; Saramaki, T.; Mitra, S.K.; Neuvo, Y. On properties and design of nonuniformly spaced linear arrays. *IEEE Trans. Acoust. Speech Signal Process.* **1988**, *36*, 372–380.
4. Haupt, R.L. Thinned arrays using genetic algorithms. *IEEE Trans. Antennas Propag.* **1994**, *42*, 993–999.
5. Yan, K.K.; Lu, Y. Sidelobe reduction in array-pattern synthesis using genetic algorithm. *IEEE Trans. Antennas Propag.* **1997**, *45*, 1117–1122.
6. Chen, K.; He, Z.; Han, C. Design of 2-dimensional sparse arrays using an improved genetic algorithm. In Proceedings of the IEEE Workshop on Sensor Array and Multichannel Processing, Waltham, MA, USA, 12–14 July 2006; pp. 209–213.
7. Cen, L.; Ser, W.; Yu, Z.L.; Rahardja, S. An improved genetic algorithm for aperiodic array synthesis. In Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, Las Vegas, NV, USA, 31 March–4 April 2008; pp. 2465–2468.
8. Trucco, A.; Murino, V. Stochastic optimization of linear sparse arrays. *IEEE J. Ocean. Eng.* **1999**, *24*, 291–299.
9. Repetto, S.; Trucco, A. Designing superdirective microphone arrays with a frequency-invariant beam pattern. *IEEE Sens. J.* **2006**, *6*, 737–747.

10. Caorsi, S.; Lommi, A.; Massa, A.; Pastorino, M. Peak sidelobe level reduction with a hybrid approach based on gas and difference sets. *IEEE Trans. Antennas Propag.* **2004**, *52*, 1116–1121.
11. Oliveri, G.; Massa, A. Genetic algorithm (GA)-enhanced almost difference set (ADS)-based approach for array thinning. *IET Microwav. Antennas Propag.* **2011**, *5*, 305–315.
12. Candes, E.; Romberg, J.; Tao, T. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE Trans. Inf. Theory* **2006**, *52*, 489–509.
13. Ji, S.; Xue, Y.; Carin, L. Bayesian compressive sensing. *IEEE Trans. Signal Process.* **2008**, *56*, 2346–2356.
14. Wang, W.; Wu, R. High resolution direction of arrival (DOA) estimation based on improved orthogonal matching pursuit (OMP) algorithm by iterative local searching. *Sensors* **2013**, *13*, 11167–11183.
15. Li, L.; Zhang, W.; Li, F. The design of sparse antenna array. *CoRR* **2008**, *11*, 110–117.
16. Carin, L. On the relationship between compressive sensing and random sensor arrays. *IEEE Antennas Propag. Mag.* **2009**, *51*, 72–81.
17. Cen, L.; Ser, W.; Cen, W.; Yu, Z.L. Linear sparse array synthesis via convex optimization. In Proceedings of the IEEE International Symposium on Circuits and Systems, Paris, France, 30 May–2 June 2010; pp. 4233–4236.
18. Prisco, G.; D’Urso, M. Exploiting compressive sensing theory in the design of sparse arrays. In Proceedings of the IEEE Radar Conference, Kansas City, MO, USA, 23–27 May 2011; pp. 865–867.
19. Oliveri, G.; Carlin, M.; Massa, A. Complex-weight sparse linear array synthesis by bayesian compressive sampling. *IEEE Trans. Antennas Propag.* **2012**, *60*, 2309–2326.
20. Fuchs, B. Synthesis of sparse arrays with focused or shaped beampattern via sequential convex optimizations. *IEEE Trans. Antennas Propag.* **2012**, *60*, 3499–3503.
21. Prisco, G.; D’Urso, M. Maximally sparse arrays via sequential convex optimizations. *IEEE Antennas Wirel. Propag. Lett.* **2012**, *11*, 192–195.
22. Candes, E.J.; Wakin, M.B.; Boyd, S.P. Enhancing sparsity by reweighted  $l_1$  minimization. *J. Fourier Anal. Appl.* **2008**, *14*, 877–905.
23. Williams, D.B.; Madisetti, V.; (Eds.) *Digital Signal Processing Handbook*, 1st ed.; CRC Press, Inc.: Boca Raton, FL, USA, 1997.
24. Zhong, X.; Premkumar, A.; Wang, H. Multiple wideband acoustic source tracking in 3-D space using a distributed acoustic vector sensor array. *IEEE Sens. J.* **2014**, *14*, 2502–2513.
25. Zheng, G. A novel spatially spread electromagnetic vector sensor for high-accuracy 2-D DOA estimation. *Multidimens. Syst. Signal Process.* **2015**, *16*, 1–26.
26. Wang, Y.; Xia, W.; He, Z.; Sun, Y. Polarimetric detection for vector-sensor processing in quaternion proper gaussian noises. *Multidimens. Syst. Signal Process.* **2015**, 1–22.
27. Compton, R.T. The tripole antenna: An adaptive array with full polarization flexibility. *IEEE Trans. Antennas Propag.* **1981**, *29*, 944–952.
28. Leprettre, B.; Martin, N.; Glangeaud, F.; Navarre, J.P. Three-component signal recognition using time, time-frequency, and polarization information-application to seismic detection of avalanches. *IEEE Trans. Signal Process.* **1998**, *46*, 83–102.
29. Bansal, R. Tripole to the rescue [antennas]. *IEEE Antennas Propag. Mag.* **2001**, *43*, 106–107.
30. Wong, K. Direction finding/polarization estimation-dipole and/or loop triad(s). *IEEE Trans. Aerosp. Electron. Syst.* **2001**, *37*, 679–684.
31. Wong, K. Blind beamforming/geolocation for wideband-FFHs with unknown hop-sequences. *IEEE Trans. Aerospace Electron. Syst.* **2001**, *37*, 65–76.
32. Lundback, J.; Nordebo, S. Analysis of a tripole array for polarization and direction of arrival estimation. In Proceedings of the IEEE Workshop on Sensor Array and Multichannel Signal Processing, Barcelona, Spain, 18–21 July 2004; pp. 284–288.
33. Au-Yeung, C.K.; Wong, K. CRB: sinusoid-sources’ estimation using collocated dipoles/loops. *IEEE Trans. Aerospace Electron. Syst.* **2009**, *45*, 94–109.
34. Yuan, X.; Wong, K.; Xu, Z.; Agrawal, K. Various compositions to form a triad of collocated dipoles/loops, for direction finding and polarization estimation. *IEEE Sens. J.* **2012**, *12*, 1763–1771.
35. Getu, B. Theoretical capacity of tripole array MIMO systems within a fixed area. In Proceedings of the Wireless and Mobile Networking Conference (WMNC), Dubai, UAE, 23–25 April 2013; pp. 1–4.
36. Zoltowski, M.; Wong, K. Closed-form eigenstructure-based direction finding using arbitrary but identical subarrays on a sparse uniform cartesian array grid. *IEEE Trans. Signal Process.* **2000**, *48*, 2205–2210.

37. Zoltowski, M.; Wong, K. ESPRIT-based 2-D direction finding with a sparse uniform array of electromagnetic vector sensors. *IEEE Trans. Signal Process.* **2000**, *48*, 2195–2204.
38. Wong, K.; Zoltowski, M. Closed-form direction finding and polarization estimation with arbitrarily spaced electromagnetic vector-sensors at unknown locations. *IEEE Trans. Antennas Propag.* **2000**, *48*, 671–681.
39. Wong, K.; Li, L.; Zoltowski, M. Root-MUSIC-based direction-finding and polarization estimation using diversely polarized possibly colocated antennas. *IEEE Antennas Wirel. Propag. Lett.* **2004**, *3*, 129–132.
40. Wong, K.; Yuan, X. Vector cross-product direction-finding with an electromagnetic vector-sensor of six orthogonally oriented but spatially noncollocating dipoles/loops. *IEEE Trans. Signal Process.* **2011**, *59*, 160–171.
41. He, J.; Ahmad, M.O.; Swamy, M. Extended-aperture angle-range estimation of multiple-fresnel-region sources with a linear tripole array using cumulants. *Signal Process.* **2012**, *92*, 939–953.
42. Zheng, G.; Wu, B.; Ma, Y.; Chen, B. Direction of arrival estimation with a sparse uniform array of orthogonally oriented and spatially separated dipole-triads. *IET Radar Sonar Navig.* **2014**, *8*, 885–894.
43. Song, Y.; Yuan, X.; Wong, K. Corrections to vector cross-product direction-finding with an electromagnetic vector-sensor of six orthogonally oriented but spatially noncollocating dipoles/loops [jan 11 160-171]. *IEEE Trans. Signal Process.* **2014**, *62*, 1028–1030.
44. Hawes, M.; Liu, W. Location optimization of robust sparse antenna arrays with physical size constraint. *IEEE Antennas Wirel. Propag. Lett.* **2012**, *11*, 1303–1306.
45. Hawes, M.; Liu, W. Sparse array design for wideband beamforming with reduced complexity in tapped delay-lines. *IEEE Trans. Acoust. Speech Lang. Process.* **2014**, *22*, 1236–1247.
46. Hawes, M.; Liu, W. Compressive sensing-based approach to the design of linear robust sparse antenna arrays with physical size constraint. *IET Microwav. Antennas Propag.* **2014**, *8*, 736–746.
47. Hawes, M.; Liu, W. Design of fixed beamformers based on vector sensor arrays. *Int. J. Antennas Propag.* **2015**, *2015*, 1–9.
48. Zhang, X.; Liu, Z.; Liu, W.; Xu, Y. Quasi-vector-cross-product based direction finding algorithm with a spatially stretched tripole. In Proceedings of the 2013 IEEE International Conference of IEEE Region 10 (TENCON13), Xi'an, China, 22–25 October 2013; pp. 1–4.
49. Research, C. CVX: Matlab Software for Disciplined Convex Programming, Version 2.0 Beta. Available online: <http://cvxr.com/cvx> (accessed on 20 September 2015).
50. Grant, M.; Boyd, S. Graph implementations for nonsmooth convex programs. In *Recent Advances in Learning and Control (a tribute to M. Vidyasagar)*; Blondel, V., Boyd, S., Kimura, H., Eds.; Springer-Verlag Limited: Berlin Heidelberg, Germany, 2008; pp. 95–110.



© 2015 by the authors; licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons by Attribution (CC-BY) license (<http://creativecommons.org/licenses/by/4.0/>).