ADAPTIVE BEAMPATTERN SYNTHESIS WITH ANTENNA SELECTION BY ITERATIVE SHRINKAGE CONTINUATION METHOD

Xiangrong Wang* *Elias Aboutanios*†

School of Electrical Engineering and Telecommunication
University of New South Wales, Sydney, Australia 2052
E-mail: * x.r.wang@unsw.edu.au; † elias@unsw.edu.au

ABSTRACT

Beampattern synthesis with antenna selection is cast as linear inverse problem (LIP) in this paper and an iterative shrinkage based optimization method is proposed to solve this problem effectively for large antenna array. The proposed method can be used to synthesize arbitrary shaped beampattern, including multibeam forming and radiation suppression within several angular regions. Since the iterative shrinkage method converges much faster compared with other considered methods, null positions can be changed adaptively according to real-time scenarios. In order to balance the trade-off between sidelobe level and null depth, the beampattern synthesis error is reweighted by an emphasis vector. A continuation strategy is also utilised to accelerate convergence rate of the proposed algorithm and the number of selected antennas is controllable by tuning the trade-off parameter value either forward or backward during iteration progress. Numerical results assess efficiency and practicality of the proposed approach for designing non-uniformly spaced arrays adaptively of up to a few hundred antennas.

Index Terms—
Adaptive beampattern synthesis, Antenna selection, Continuation strategy, Fast convergence rate, Iterative shrinkage

I. INTRODUCTION

For beampattern synthesis, antenna selection is an economical and practical method to use with respect to minimizing the weight, reducing hardware cost and complexity of the array. There exist many effective methods of synthesizing sparse array in the literature, such as heuristic method [1], convex optimization [2], Bayesian Compressive Sensing [3], Matrix Pencil [4] and iterative fast Fourier transform (FFT) algorithm [5] and so on. However most of these methods are considered computationally expensive especially when the problem becomes large scale. Thus these methods cannot be utilised to adaptively synthesize beampattern according to real-time scenarios. Moreover the number of selected antennas is not predictable and controllable for most of the methods which has increased the hardware cost by adding more front-ends in the receiver.

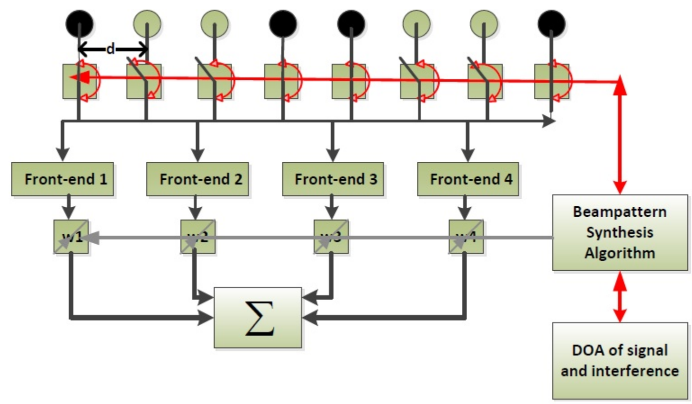

Fig. 1. Block diagram of adaptive antenna selection strategy

In order to synthesize shaped beampattern with multibeam and arbitrary null positions adaptively, a fast beampattern synthesis algorithm based on antenna selection is proposed in this paper. The block diagram of this strategy is shown in Fig. 1. There are only K_{max} (K_{max} is less than the number of antennas, N) front-ends installed in the receiver and thus the hardware cost is reduced dramatically. The beampattern synthesis algorithm is developed based on estimated arrival directions of the desired signal and interference. The K (Here $K \leq K_{max}$ should be guaranteed) antennas with non-zero entries in the calculated excitation weight vector are switched on and connected to following front-ends with corresponding coefficients and remaining antennas are kept off. In order to reduce the computational time, we adopt an iterative shrinkage method which has been effectively used to address large-scale optimization problems in the area of compressive sensing and image processing due to its simple structure. Each iteration of this algorithm is comprising of a multiplication by a matrix and its transpose, along with a scalar shrinkage step on the obtained result [6]. An emphasis vector is introduced to balance the trade-off between sidelobe level and null depth of the synthesized beampattern. A continuation strategy is also utilized to speed up convergence and guarantee the number of selected antennas $K \leq K_{max}$ by tuning the trade-off parameter value either forward or backward during iteration progress.

This paper is organised as follows. In section II, the LIP model of array beampattern synthesis is described and the proposed Iterative Shrinkage Continuation method is introduced as well. To show its efficiency, some numerical results are shown in section III. Conclusions are drawn in section IV.

II. MATHEMATICAL MODEL

Consider an N-antenna uniform symmetric array with inter-element spacing d like shown in Fig. 1. A plane wave with wavelength λ is impinging on the array from a $\mathbf{u} = \cos\theta$ direction. We assume, both the whole array and the composed subarray are symmetric due to the symmetric synthesized beampattern, then only one half of the array needs to be considered. The selection of an antenna in the right half array implies the selection of the corresponding antenna in the left half. Let **P** denote the position coordinate vector of the right half array, then the array beampattern is defined as

$$B(\mathbf{u}) = \mathbf{w}^T cos(\frac{2\pi}{\lambda}\mathbf{P}\mathbf{u}). \tag{1}$$

where $\mathbf{w} \in R^N$ is the array weight vector and ^{T} denotes transpose operation. In order to transform the array beampattern synthesis problem into LIP, we stack L uniformly spaced desired beampattern samples into a vector $\mathbf{f}_d \in R^L$ ($L \ll N$). Define the transformation matrix as $\mathbf{A} \in R^{L\times N}$ with the i_{th} row $\mathbf{A}_i$ as

$$\mathbf{A}_i^T = cos(\frac{2\pi}{\lambda}\mathbf{P}\mathbf{u}_i)\ i = 1, ..., L \tag{2}$$

We can see that the transformation matrix $\mathbf{A}^T$ is orthogonal and the special structure of **A** makes the proposed method suitable for solving the sparse array synthesis problem [7], which is formulated as composing a subarray with as few as possible antennas in terms of synthesizing the desired beampattern $\mathbf{f}_d$, i.e.

$$\begin{aligned} \min \quad & \| \mathbf{w} \|_0 \\ \text{s.j.t} \quad & \mathbf{A}\mathbf{w} = \mathbf{f}_d \end{aligned} \tag{3}$$

where the l_0 norm function $\| \mathbf{w} \|_0 = | i : w_i \neq 0 |$ is defined as the number of nonzero entries of the weight vector **w**. If $w_i = 0$, then the i_{th} antenna is discarded; otherwise the nonzero value is taken as the corresponding weight value. However, l_0 norm minimization problem is non-convex and generally very hard to solve, as their solution usually requires an intractable combinatorial search [8]. A common alternative approach is to relax Eq. (3) by replacing the l_0 norm with the well-known l_1 norm, which has somewhat similar "sparsity properties" [8]. The new problem,

$$\begin{aligned} \min \quad & \| \mathbf{w} \|_1 \\ \text{s.j.t} \quad & \mathbf{A}\mathbf{w} = \mathbf{f}_d \end{aligned} \tag{4}$$

called Basis Pursuit [9], is convex. Since the synthesized beampattern can not be exactly same with the desired one, there is inevitably synthesis error noise in certain scenarios. Therefore solving the unconstrained l_1-norm based least square problem

$$\min_{\mathbf{w}\in R^N} \{ \| \mathbf{w} \|_1 + \mu\frac{1}{2} \| \mathbf{A}\mathbf{w} - \mathbf{f}_d \|^2 \} \tag{5}$$

is more preferable than solving the constrained problem Eq. (4). In order to balance the trade-off between sidelobe level and null depth, an emphasis parameter $\boldsymbol{\alpha}$ is adopted to reweight the synthesis beampattern error as follows:

$$\min_{\mathbf{w}\in R^N} \{ \| \mathbf{w} \|_1 + \mu\frac{1}{2} \| \boldsymbol{\alpha} \odot (\mathbf{A}\mathbf{w} - \mathbf{f}_d) \|^2 \} \tag{6}$$

Where $\odot$ denotes the Hadamard Product. Use function $H(\mathbf{w}) : R^N \to R$ to represent the latter part of Eq. (5) and then the l_1 regularization problem is rewritten as

$$\min_{\mathbf{w}\in R^N} \{ \| \mathbf{w} \|_1 + \mu H(\mathbf{w}) \} \tag{7}$$

where μ is the trade-off parameter that balances between the solution sparseness and the minimization of synthesized beampattern error. Generally, a smaller parameter μ yields a sparser minimizer $\mathbf{w}^*$, while also a greater discrepancy on $H(\mathbf{w})$. Let us define the emphasis vector $\boldsymbol{\alpha}$ with constant l_2 norm $\sqrt{L}$ and then a larger entry of $\boldsymbol{\alpha}$ implies more emphasis, for example a deeper null can be realised by imposing a larger weight. Let $\nabla H(\mathbf{w})$ be the gradient of $H(\mathbf{w})$, i.e.

$$\nabla H(\mathbf{w}) = \mathbf{A}^T(\boldsymbol{\alpha} \odot \boldsymbol{\alpha} \odot (\mathbf{A}\mathbf{w} - \mathbf{f}_d)) \tag{8}$$

Since l_1 norm function is not differentiable, its sub-gradient is calculated as follows:

$$SIGN(\mathbf{w})_i = \{\partial \| \mathbf{w} \|_1\}_i = \begin{cases} \{1\} & w_i > 0, \\ [-1, 1] & w_i = 0, \\ \{-1\} & w_i < 0 \end{cases} \tag{9}$$

It is well-known from convex analysis [9] that the optimality condition for $\mathbf{w}^*$ being one of the optimal solutions of Eq. (7) is

$$\mathbf{0} \in SIGN(\mathbf{w}^*) + \mu\nabla H(\mathbf{w}^*) \tag{10}$$

where $\mathbf{0}$ is the zero vector in R^N. Combining Eq. (9) and Eq. (10), we can get that

$$\mu\nabla H_i(\mathbf{w}^*) \begin{cases} = -1 & w_i^* > 0, \\ \in [-1, 1] & w_i^* = 0, \\ = 1 & w_i^* < 0, \end{cases} \tag{11}$$

We can derive another equivalent optimality condition for Eq. (7) based on Eq. (11) [7]. That is , for any scalar $\tau > 0$,

$$w_i^* = s_{\frac{\tau}{\mu}}(h(\mathbf{w}^*)_i) \tag{12}$$

with $h(\mathbf{w}^*) = \mathbf{w}^* - \tau\nabla H(\mathbf{w}^*)$. We refer to $s_\rho(\cdot)$ as the shrinkage operator as follows,

$$s_\rho(\cdot) = \text{sign}(\cdot)\max\{| \cdot | - \rho, 0\}, where\ \rho > 0 \tag{13}$$

Table I. Iterative Shrinkage Continuation Algorithm

Step1	Initialising $\mathbf{w}_0 = \mathbf{0}$, $\mu = 0.5$, $\tau = 1/\lambda_{max}$ and uniform $\boldsymbol{\alpha}$;

Step2	While (not converged, i.e. $e_{k+1} \geq 1e-4$)
	Iterate according to Eq. (14)
	$e_{k+1} = \parallel \mathbf{w}_{k+1} - \mathbf{w}_k \parallel_2$
	End while

Step3	If the desired beampattern achieved, then terminate; Otherwise, set $\mathbf{w}_0 = \mathbf{w}_{k+1}$ and go to Step 4a or 4b.
Step4a	Change value of μ continuously, and iterate Step 2-Step 3, until the suitable value of μ is found.
Step4b	Fix the chosen value of μ and continuously change value of null weight in $\boldsymbol{\alpha}$; Iterate Step 2-Step 3.

where $\text{sign}(x)$ is the indicator function with value zero when $x = 0$. Intuitively, $h(\cdot)$ resembles a gradient descendent step for decreasing $H(\mathbf{w})$, and $s_\rho(\cdot)$ reduces the magnitude of each nonzero component of the input vector by an amount less than or equal to ρ, thus decreasing the l_1 norm [7]. It is natural to get the unique minimizer solution of Eq. (7) using an iterative method, i.e. the solution of the $k+1_{th}$ iteration is

$$\mathbf{w}_{k+1} = s_{\frac{\tau}{\mu}}(\mathbf{w}_k - \tau \nabla H(\mathbf{w}_k)) \tag{14}$$

The convergence rate of the iterative shrinkage method is q-linear under certain conditions on H and τ. Under weaker conditions, they also exhibits r-linear convergence rate. The detail of convergence analysis can be found in [7]. Like what we have mentioned above, the orthogonal property of transformation matrix $\mathbf{A}$ makes the proposed algorithm more effective for array beampattern synthesis problem. According to [7], the parameter τ should be chosen within the range $\tau \in (0, 2/\lambda_{max})$ in order to satisfy the convergence condition, where λ_{max} is the maximum eigenvalue of the Hessian of $H(\mathbf{w})$. Recently some speed-up algorithms are shown to emerge from different considerations, such as TwIST is proposed to solve the problem with ill-conditioned matrix $\mathbf{A}$ [10]. Here we adopt the continuation strategy to increase the value of parameter μ continuously [7]. The approximate solution of the problem associated with μ_p is used as the starting point of the optimization problem associated with μ_{p+1}. The detail procedure of Iterative Shrinkage Continuation method is shown in Table I.

III. SIMULATION RESULTS

This section exhibits some numerical results using the proposed method for both linear and planar array to validate the effectiveness and efficiency.

III-A. Linear Array

Firstly the effect of trade-off parameter μ on the array sparseness and synthesized beampattern is investigated. Let us assume a linear symmetric array with 200 uniformly spaced antennas and a uniform emphasis vector. The desired beampattern has one mainlobe around $\theta = 0$ and two nulls within the elevation angle $(30-32)$ and $(59-61)$ degrees respectively with $-80dB$ depth. The desired sidelobe level

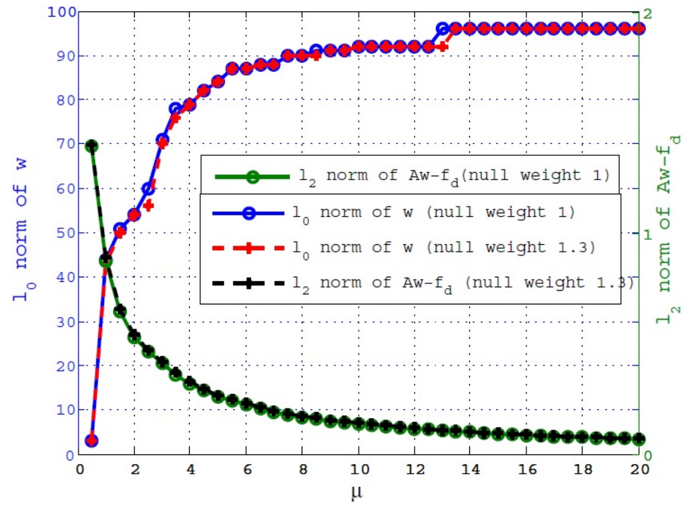

Fig. 2. Trade-off curve with respect to different μ and null weight.

(SLL) is set to be $-40dB$. The number of beampattern samples, L, is chosen as 100 in this example. The value of the trade-off parameter μ is changing from 0.5 to 20. Since there are only $K_{max} = 150$ front-ends installed in the receiver, the suitable number of selected antennas needs to be determined first. The trade-off curve between the l_0 norm of solution $\mathbf{w}$ and the synthesis beampattern error is shown in Fig. 2. We can see that the simulation result coincides exactly with what we have predicted mathematically: the sparseness of solution $\mathbf{w}$ decreases with the increasing value of μ, while the synthesized beampattern becomes closer to the desired one. Moreover it is convenient to control the number of selected antennas, K less than or equal to K_{max}, by tuning the value of μ either forward or backward during iteration progress through the proposed continuation strategy.

Here the trade-off parameter is chosen as $\mu = 2$ and $\mu = 3.5$, correspondingly the number of selected antennas is $K = 54$ and $K = 60$ respectively. The synthesized beampatterns of two arrays are compared in Fig. 3. We can see that the null depth is deeper and the sidelobe level is also a little lower for the larger array than the smaller one. However the performance loss can be compensated by tuning the emphasis vector as shown in the next experiment. It is also noted that the mainlobe width does not change much even with larger aperture length, because the number of beampattern samples L is usually less than $N/2$ and thus the minimum mainlobe width is restricted to approximately $180/N$ degrees for the desired beampattern using the proposed method.

Next the effect of emphasis vector on the synthesized beampattern shape is discussed. Since there is not much chance to decrease the mainlobe width (unless changing the beampattern sample number L with the sacrifice of high sidelobes) and interference suppression is the main focus of

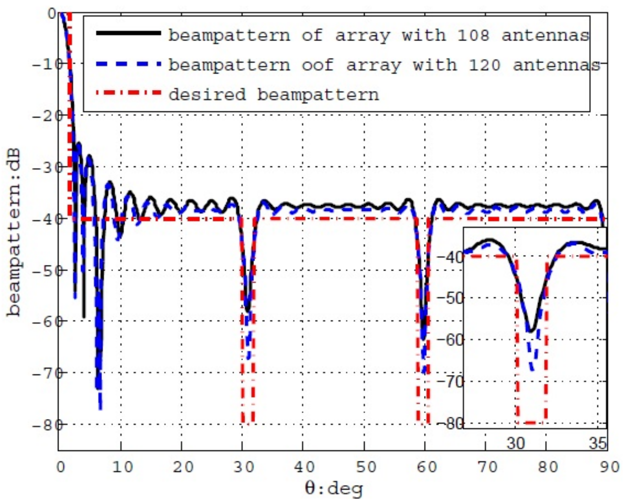

Fig. 3. Beampatterns of two arrays with different numbers of selected antennas (Uniform emphasis vector is chosen here)

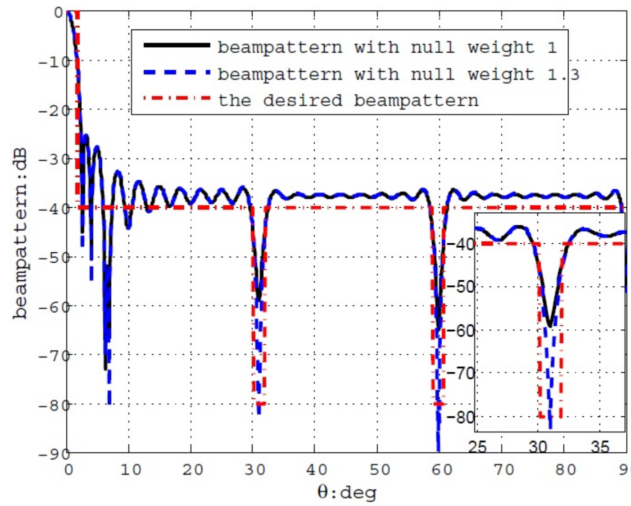

Fig. 4. Beampatterns of 108-antenna subarray with different null weights (The trade-off parameter μ is 2 here)

our proposed adaptive antenna selection strategy, the weights of emphasis vector within mainlobe and sidelobe regions are set to be one. The null weight value is changing from 0 to 3 in order to see the effect of different null weights. The trade-off parameter μ is set to be 2 according to the above experiment. The trade-off curve with respect to different null weights is shown in Fig. 2. We can see that the trade-off curve does not change much with respect to different null weights. Thus we determine the value of μ first and then adjust the null weight according to the desired beampattern in the algorithm Table I. The beampatterns of null weight value being 1 and 1.3 are compared in Fig. 4. it is clear that increasing null weight makes the sidelobe level a little higher, but with much deeper nulls as what we have expected from Eq. (7). Moreover we can also see that the synthesized beampattern of the smaller array with 108 antennas and null weight being 1.3 has deeper nulls than the larger array with 120 antennas and null weight being 1 by comparing Fig. 3 and Fig. 4. Thus the selected subarray can preserve the array processing performance with dramatic hardware cost reduction.

Finally the computational time of the proposed algorithm is compared with some other considered methods, including the iterative FFT and convex optimization method which uses the interior point based second order cone programming to solve Eq. (5) through CVX software. The iterative reweighted l_1 norm is adopted to obtain a sparse solution for the convex optimization method [2]. A uniform emphasis vector is chosen for this experiment. A 200-antenna symmetric linear array which is thinned to 144 elements in [5] is used for comparison. The corresponding beampatterns synthesized by convex optimization and the proposed method are compared in Fig. 5. The simulation results are summarized and compared in Table II where the "Average SLL" is referred to the average sidelobe level excluding high sidelobes around the mainlobe (Note that the results corresponding to the iterative FFT method are taken from [5]). The operating laptop for simulation has 2.5GHz clock signal and 8GB RAM memory. We can see that although the beampattern synthesized by the iterative FFT method has narrower mainlobe width, whereas with higher sidelobe level and much more computational time (of 10-second order) due to 1000 runs. The beampatterns synthesized by convex optimization and the proposed method are nearly same, however the proposed method is much faster which is more important for real-time application. The excitation weight vector obtained from the proposed method is shown in the lower plot of Fig. 5.

Table II. Comparison of Simulation Results Among Shrinkage, Convex optimization and iterative FFT methods

Method	3-dB Beamwidth (degree)	Peak SLL (dB)	Computational time (second)
Shrinkage	1	-25.56	2e-4
CVX	0.9	-24.85	1.5185
FFT	0.6	-24.64	10-second order

Method	Average SLL (dB)	Aperture Length (λ)	Antenna Number (selected)
Shrinkage	-39	99.5	144
CVX	-40.5	99.5	144
FFT	-24.64	99.5	144

III-B. Planar Array

Now we proceed to run our simulation with a 20×20 symmetric planar array. We choose the trade-off parameter as $\mu = 2$ and correspondingly a 236-antenna symmetric subarray is composed. The synthesized beampattern is shown in Fig. 6. We can see that the null depth is $-80dB$. The computational time for synthesizing this planar array is at

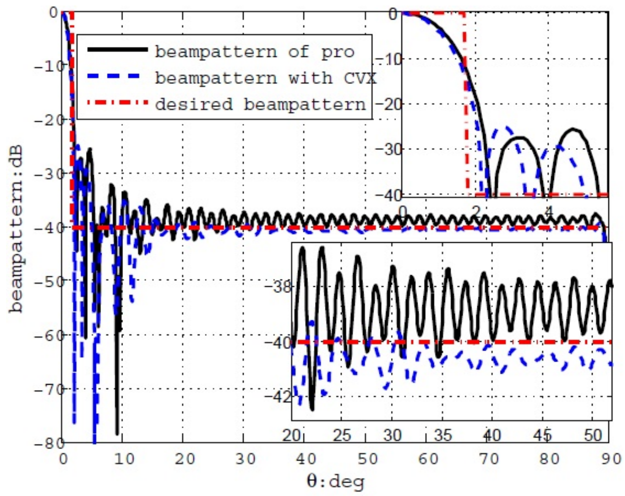

Fig. 5. Beampatterns of convex optimization and iterative shrinkage method. The inside plot is the excitation weight returned from iterative shrinkage method.

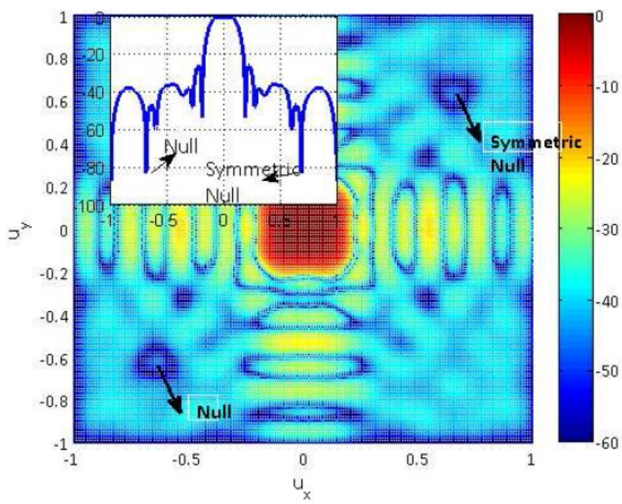

Fig. 6. Beampattern of 236-antenna subarray with Iterative Shrinkage Continuation method, the inside plot is the cut along the diagonal line

most 4.3e-3 seconds. In one word, the Iterative Shrinkage Continuation method works well for both linear and planar array. The fast convergence property of the proposed approach is more evident for synthesizing large scale antenna array.

IV. CONCLUSION

An Iterative Shrinkage Continuation method is proposed in this paper to synthesize sparse arrays with shaped beampattern adaptively. The proposed method can design arbitrary shaped beampattern with multi-beam and several null angle regions. The main advantage of the proposed method is its fast convergence rate compared with the classical steepest descent method and other considered sparse array synthesis algorithms especially for large antenna array. Moreover the trade-off between sidelobe level and null depth can be balanced by adjusting the emphasis vector and the number of selected antennas is controllable by tuning the trade-off parameter during iteration progress using the continuation strategy. Therefore, the proposed antenna selection based Iterative Shrinkage Continuation method can be used in real-time application to adaptively synthesize large array beampattern.

V. REFERENCES

[1] L. Cen, Z. Yu, W. Ser, and W. Cen, “Linear aperiodic array synthesis using an improved genetic algorithm,” *Antennas and Propagation, IEEE Transactions on*, vol. 60, no. 2, pp. 895–902, 2012.

[2] S. Nai, W. Ser, Z. Yu, and H. Chen, “Beampattern synthesis for linear and planar arrays with antenna selection by convex optimization,” *Antennas and Propagation, IEEE Transactions on*, vol. 58, no. 12, pp. 3923–3930, 2010.

[3] G. Oliveri, P. Rocca, and A. Massa, “Reliable diagnosis of large linear arrays:a bayesian compressive sensing approach,” *Antennas and Propagation, IEEE Transactions on*, vol. 60, pp. 4627 –4636, oct.

[4] Y. Liu, Z. Nie, and Q. Liu, “Reducing the number of elements in a linear antenna array by the matrix pencil method,” *Antennas and Propagation, IEEE Transactions on*, vol. 56, no. 9, pp. 2955–2962, 2008.

[5] W. Du Plessis, “Weighted thinned linear array design with the iterative fft technique,” *Antennas and Propagation, IEEE Transactions on*, vol. 59, no. 9, pp. 3473–3477, 2011.

[6] M. Elad, *Sparse and redundant representations: from theory to applications in signal and image processing*. Springer, 2010.

[7] E. Hale, W. Yin, and Y. Zhang, “A fixed-point continuation method for l1-regularized minimization with applications to compressed sensing,” *CAAM TR07-07, Rice University*, 2007.

[8] E. Treister and I. Yavneh, “A multilevel iterated-shrinkage approach to l 1 penalized least-squares minimization,” *Signal Processing, IEEE Transactions on*, vol. 60, no. 12, pp. 6319–6329, 2012.

[9] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2004.

[10] J. Bioucas-Dias and M. Figueiredo, “A new twist: two-step iterative shrinkage/thresholding algorithms for image restoration,” *Image Processing, IEEE Transactions on*, vol. 16, no. 12, pp. 2992–3004, 2007.