Efficient Optimization of Reconfigurable Parasitic Antenna Arrays Using Geometrical Considerations

Prabhat Baniya
University of Massachusetts Dartmouth
North Dartmouth, MA 02747-2300, USA
E-mail: pbaniya@umassd.edu

Samee Ur Rehman and Jon W. Wallace
Jacobs University Bremen
28759 Bremen, Germany
E-mail: s.rehman@jacobs-university.de, wall@ieee.org

***Abstract*—Reconfigurable aperture (RECAP) antennas can be considered a generalization of the reconfigurable antenna concept, consisting of a large array of reconfigurable elements and supporting many applications like adaptive matching, frequency agility, and beam- and null-formation. Although providing similar functions to phased arrays, the non-linear and non-convex nature of the objective function requires global search methods such as genetic algorithms or particle swarm optimization, which may be inappropriate for real-time, in-situ optimization. To overcome this difficulty, a first-order model for parasitic RECAPs is developed herein, whose simple geometric interpretation leads to straightforward and useful solutions for beamforming and nullsteering. It is also demonstrated how the first-order solution can be used to seed an efficient local optimizer to find exact solutions to the complicated full-order objective function. The utility of the method is demonstrated through numerical examples and performance is compared with a genetic algorithm.**

I. Introduction

Parasitic antenna arrays were introduced by Harrington in [1], and more recently have gained significant attention [2–4] due to their ability to perform beamforming, nullsteering, and dynamic matching by simple electronic tuning of the parasitic reactive loads. These architectures have the potential to support required smart-antenna operations with only a single radio-frequency (RF) and digital signal processing (DSP) channel, providing reduced cost and power consumption.

Unfortunately, a potential difficulty with reactively controlled parasitic arrays is that the optimization of the parasitic loads for a particular application is a non-linear and non-convex optimization problem, usually requiring global search methods [5]. Methods such as genetic algorithms (GAs), particle swarm optimization, and ant colony optimization have been successfully employed, but these methods usually require a lengthy search of the domain space, making them inappropriate for real-time, in-situ optimization of the parasitic arrays. Although libraries of useful solutions for a parasitic array could be generated off-line and stored for real-time use, such methods do not allow the system to adapt to new unforeseen conditions.

In this paper, an efficient method for optimizing reconfigurable parasitic arrays is presented, allowing solutions for beamforming and nullsteering to be directly computed. The method is based on a reduced-order approximation of the response of the parasitic array, not only allowing fast solutions to be developed, but also useful insight on the operation of parasitic arrays to be developed, which is typically not gained from global optimization techniques like GAs. It is also demonstrated how the first-order solution can be used to seed a Newton-based optimization method that efficiently finds exact nulls of the original non-convex objective function.

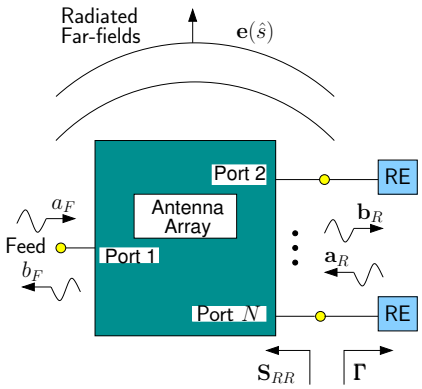

Fig. 1. A general parasitic antenna array

The paper is organized as follows: Section II briefly presents the network model and parameters of the parasitic antenna array. Section III explains the reduced-order modeling strategy, first-order solutions for beamforming and nulls, and the refinement procedure using a Newton-based method. Section IV presents numerical examples and compares performance with a genetic algorithm. Concluding remarks are given in Section V.

II. Parasitic Antenna Array

Figure 1 depicts the parameters of a generic parasitic antenna array. Although only a single polarization is considered in this work, the method is easily extended to multiple polarizations.

The depicted N-port antenna array is completely characterized by the $N\times N$ S-parameter matrix ($\mathbf{S}$) looking into the array and the N Z_0-terminated embedded radiation patterns of the elements, where $e_k(\hat{s})$ is the embedded pattern in direction $\hat{s}$ for the kth port [6]. For use as a parasitic array, we designate Port 1 as the feed with input and output waves a_F and b_F, and Ports 2 through N as parasitic elements, whose input and output waves are collected into the vectors $\mathbf{a}_R$ and $\mathbf{b}_R$. Likewise, the Z_0-terminated embedded radiation pattern of the feed is $e_{\mathrm{F}}(\hat{s})$ and radiation patterns of the parasitic ports are collected into the vector $\mathbf{e}_{\mathrm{R}}(\hat{s})$. The $N-1$ parasitic antenna

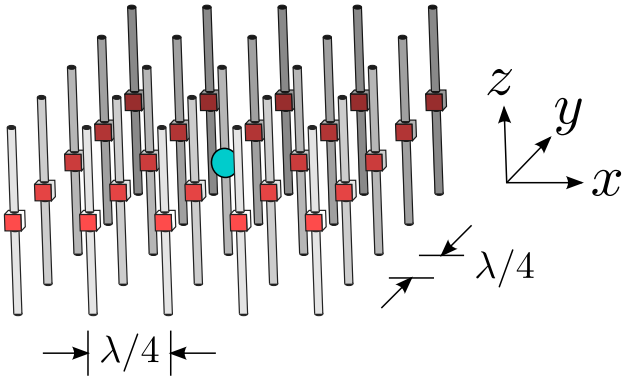

Fig. 2. 5×5 parasitic dipole array, where the center element is the active feed and surrounding elements are reactive reconfigurable loads

ports are terminated with reactive elements having a diagonal input reflection matrix $\mathbf{\Gamma}$. To simplify notation, $\mathbf{S}$ is partitioned according to

$$\mathbf{S} = \begin{bmatrix} S_{\mathrm{FF}} & \mathbf{s}_{\mathrm{FR}} \\ \mathbf{s}_{\mathrm{RF}} & \mathbf{S}_{\mathrm{RR}} \end{bmatrix}. \tag{1}$$

Network analysis reveals that the radiated far-fields for a single polarization of the general parasitic array into direction $\hat{s}$ is [6]

$$e(\hat{s}) = \left[e_{\mathrm{F}}(\hat{s}) + \mathbf{e}_{\mathrm{R}}^{T}(\hat{s})\mathbf{\Gamma}(\mathbf{I} - \mathbf{S}_{\mathrm{RR}}\mathbf{\Gamma})^{-1}\mathbf{s}_{\mathrm{RF}}\right] a_{\mathrm{F}}. \tag{2}$$

The goal of this work is then to find efficient direct solutions for $\mathbf{\Gamma}$ that maximize gain (beamforming) or minimize gain (null steering) for arbitrary target directions.

The specific parasitic array that will be considered in Section IV is depicted in Figure 2, consisting of 5×5 dipoles with $\lambda/4$ inter-element spacing, where λ is the free-space wavelength, similar to the array presented in [7]. The center dipole is the feed and the other dipoles are terminated with lossless reactive loads. S-parameters and embedded patterns of the dipoles are found using the Numerical Electromagnetics Code (NEC) assuming dipole length 0.475λ and radius 0.001λ with 21 wire segments per dipole.

III. Reduced-Order Parasitic RECAP Model

A basic problem with minimizing or maximizing radiated power given by $|e(\hat{s})|^2$ based on (2), is the non-convex dependence on $\mathbf{\Gamma}$, which leads to many local minima or maxima and typically necessitates a global search procedure such as a GA. This section describes a simple reduced-order representation of a parasitic array that not only allows simple direct optimization, but also provides valuable insight on the operation of the array.

Consider a parasitic array with $a_{\mathrm{F}} = 1$ and low mismatch ($\mathbf{S}_{\mathrm{RR}} \approx 0$) of the parasitic antennas, such that the inverse term in (2) representing multiple reflections among the REs can be treated as identity. This approximation leads to a first-order model where the radiation pattern depends linearly on the loads, and considering a single radiation direction $\hat{s}$, the

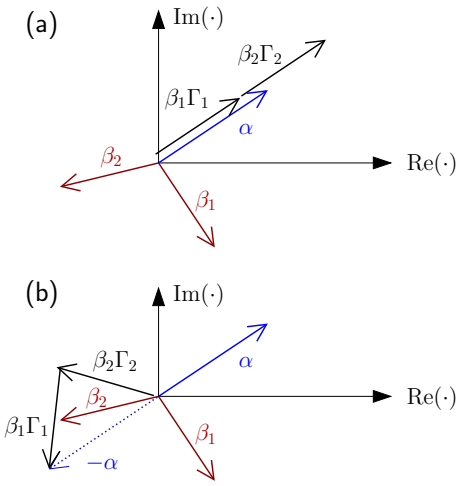

Fig. 3. Geometrical interpretation of vector alignment for first-order (a) beamforming and (b) nullsteering

first-order model is

$$e(\hat{s}) = \underbrace{e_{\mathrm{F}}(\hat{s})}_{\alpha} + \sum_{m=1}^{N-1} \underbrace{e_{\mathrm{R},m}(\hat{s}) s_{\mathrm{RF},m}}_{\beta_m} \Gamma_m, \tag{3}$$

where we define $\Gamma_m = \Gamma_{mm}$. Visualizing the phasors α, β_m, and Γ_m in the complex plane, there is a simple geometric interpretation for the dependence of radiated field in direction $\hat{s}$ on the load reflection coefficients Γ_m as illustrated in Figure 3. Assuming losses reconfigurable elements ($|\Gamma_m| = 1$), changing the phases of the Γ_m simply rotates the vectors β_m in the complex plane, and the sum of all the rotated vectors gives the radiated field due to the reconfigurable elements. Figures 3(a) and (b) illustrate how the vectors can be aligned for beamforming or nullsteering for the simple case of a 2-element RECAP, and the general procedure is explained below.

A. Beamforming Solution

Beamforming is perhaps the simplest application of the RECAP, and the maximization of (3) is accomplished by choosing the phases of Γ_m to align the phases of the β_m with the phase of α. This ensures that all waves from the REs in the direction $\hat{s}$ will add coherently with the radiated wave from the feed. The general solution is $\Gamma_m = \exp[j(\angle\alpha - \angle\beta_m)]$. Figure 3(a) illustrates the principle for a single feed and two REs.

The geometrical solution also yields insight on the gain limitation of the RECAP, since the magnitude of the field in direction $\hat{s}$ has a maximum value of $|\alpha| + \sum_n |\beta_n|$.

B. Nullsteering Solution

Consider now creating a single null in direction $\hat{s}$. In this case, the phases of the Γ_m should be chosen to cancel the wave due to the feed in direction $\hat{s}$. However, care should be taken since simply aligning the vectors in direction $-\alpha$ may lead to overshoot. Figure 3(b) depicts an arrangement for two REs that will generate a null. For a large number of reconfigurable elements, the best strategy for creating a null is not obvious,

since there may be many possible vector-alignment solutions, and parameterizing all of these solutions appears to be non-trivial.

We propose a simple heuristic approach for vector-alignment for null formation, where the β_m are first sorted according to descending magnitude. The phases of the loads are then computed sequentially, where at step m, we have

$$\angle\Gamma_m = \text{angle}(-\alpha - \sum_{n=1}^{m-1} \beta_n \Gamma_n) - \beta_m. \tag{4}$$

The basic idea is to use the longest β_m vectors first to move coarsely in the direction of the goal. As m increases and we are progressively closer to the goal, shorter β_m vectors should be used to allow movements in smaller increments.

Note that like beamforming, the geometric interpretation lends insight on the limits of the parasitic array for the nullsteering application. A perfect null can be created in a single direction as long as $\sum_n |\beta_n| \leq |\alpha|$, which indicates the minimum number of elements that are needed for null formation.

C. Multiple Goals

For real applications, we will typically have multiple goals, such as the formation of multiple nulls or a null in one direction and high gain in another direction. The geometric interpretation of the first-order expression still holds for K goals, where the kth goal is concerned with direction $\hat{s}_k$, and

$$e(\hat{s}_k) = \underbrace{e_{\text{F}}(\hat{s}_k)}_{\alpha_k} + \sum_{m=1}^{N-1} \underbrace{e_{\text{R},m}(\hat{s}_k) s_{\text{RF},m}}_{\beta_{km}} \Gamma_m. \tag{5}$$

Now, however, we have K plots like Figure 3, and the single set of Γ_m needs to be chosen to rotate β_{km} in each plot to enhance (for beamforming) or cancel (for nulling) the α_k on those plots. Obviously, we usually have competing goals where optimal alignment for one goal will move in sub-optimal directions for other goals.

Developing methods for satisfying joint goals on the basis of the first-order model appears to be difficult, especially for nulls that are sensitive to very small errors. Since beamforming is mostly insensitive to error, in this paper we only consider using the first-order model for beamforming and use an efficient Newton-based optimizer to modify that solution to find one or more nulls. However, methods for directly exploiting reduced-order models to find useful solutions for multiple goals remains an important aim of this research.

D. Newton-Based Local Optimization

An obvious drawback of the first-order solution is that multiple reflections cause (3) to not be precisely equal to (2). Although beamforming is somewhat insensitive to this effect, even small amounts of error can destroy null formation. Here we provide a local optimization approach based on Newton's method that allows exact nulls to be found in an efficient manner.

Our goal is to develop a first-order approximation of (2) within the neighborhood of the reflection coefficient matrix $\mathbf{\Gamma}$, which can be accomplished using a multidimensional Taylor series. Considering lossless reconfigurable elements, $\Gamma_{\ell\ell} = \exp(j\theta_\ell)$, and using the identities for matrix derivatives

$$\frac{\partial \mathbf{A}^{-1}}{\partial x} = -\mathbf{A}^{-1} \frac{\partial \mathbf{A}}{\partial x} \mathbf{A}^{-1}, \tag{6}$$

$$\frac{\partial \mathbf{a}^T \mathbf{A} \mathbf{b}}{\partial x} = \mathbf{a}^T \frac{\partial \mathbf{A}}{\partial x} \mathbf{b}, \tag{7}$$

the derivative of (2) with respect to the phase of the ℓth load can be evaluated as

$$\frac{\partial e(\hat{s})}{\partial \theta_\ell} = -\mathbf{e}_{\text{R}}^T (\mathbf{\Gamma}^{-1} - \mathbf{S}_{\text{RR}})^{-1} \frac{\partial (\mathbf{\Gamma}^{-1} - \mathbf{S}_{\text{RR}})}{\partial \theta_\ell} (\mathbf{\Gamma}^{-1} - \mathbf{S}_{\text{RR}})^{-1} \mathbf{s}_{\text{RF}}. \tag{8}$$

Evaluating the inner derivative in (8),

$$\left[\frac{\partial (\mathbf{\Gamma}^{-1} - \mathbf{S}_{\text{RR}})}{\partial \theta_\ell} \right]_{mn} = \frac{\partial (\mathbf{\Gamma}^{-1} - \mathbf{S}_{\text{RR}})_{mn}}{\partial \theta_\ell} \tag{9}$$

$$= -j \underbrace{\delta_{m\ell} \delta_{n\ell}}_{(\mathbf{1}_{\ell\ell})_{mn}} \exp(-j\theta_\ell) \tag{10}$$

where $\mathbf{1}_{ik}$ is an elementary matrix that is all zeros except for a 1 for the ikth element. Combining (8) and (10),

$$\frac{\partial e(\hat{s})}{\partial \theta_\ell} = j \mathbf{e}_{\text{R}}^T (\hat{s}) (\mathbf{\Gamma}^{-1} - \mathbf{S}_{\text{RR}})^{-1} \mathbf{1}_{\ell\ell} (\mathbf{\Gamma}^{-1} - \mathbf{S}_{\text{RR}})^{-1} \mathbf{s}_{\text{RF}} e^{-j\theta_\ell}, \tag{11}$$

and the $(N-1)\times 1$ gradient vector $\mathbf{d}$ is formed with elements $d_\ell = \partial e(\hat{s}) / \partial \theta_\ell$. Next, the first-order Taylor series for real and imaginary parts of (2) is formed by expanding about the point $\boldsymbol{\theta}_n$ to obtain the solution at a new point $\boldsymbol{\theta}_{n+1}$ according to

$$e_{\text{re}}(\hat{s}, \boldsymbol{\theta}_{n+1}) = e_{\text{re}}(\hat{s}, \boldsymbol{\theta}_n) + \mathbf{d}_{\text{re}}^{\text{T}}(\boldsymbol{\theta}_n)(\boldsymbol{\theta}_{n+1} - \boldsymbol{\theta}_n), \tag{12}$$

$$e_{\text{im}}(\hat{s}, \boldsymbol{\theta}_{n+1}) = e_{\text{im}}(\hat{s}, \boldsymbol{\theta}_n) + \mathbf{d}_{\text{im}}^{\text{T}}(\boldsymbol{\theta}_n)(\boldsymbol{\theta}_{n+1} - \boldsymbol{\theta}_n), \tag{13}$$

where

$$\mathbf{d}_{\text{re}}(\boldsymbol{\theta}) = \text{Re}\left\{ \mathbf{d}|_{\mathbf{\Gamma}(\boldsymbol{\theta})} \right\} \quad e_{\text{re}}(\hat{s}, \boldsymbol{\theta}) = \text{Re}\left\{ e(\hat{s})|_{\mathbf{\Gamma}(\boldsymbol{\theta})} \right\} \tag{14}$$

$$\mathbf{d}_{\text{im}}(\boldsymbol{\theta}) = \text{Im}\left\{ \mathbf{d}|_{\mathbf{\Gamma}(\boldsymbol{\theta})} \right\} \quad e_{\text{im}}(\hat{s}, \boldsymbol{\theta}) = \text{Im}\left\{ e(\hat{s})|_{\mathbf{\Gamma}(\boldsymbol{\theta})} \right\}. \tag{15}$$

To find a null, we set $e_{\text{re}}(\hat{s}, \boldsymbol{\theta}_{n+1}) = e_{\text{im}}(\hat{s}, \boldsymbol{\theta}_{n+1}) = 0$, or

$$\underbrace{[e_{\text{re}}(\hat{s}, \boldsymbol{\theta}_n) \; e_{\text{im}}(\hat{s}, \boldsymbol{\theta}_n)]^T}_{\mathbf{b}} = -\underbrace{[\mathbf{d}_{\text{re}}(\boldsymbol{\theta}_n) \; \mathbf{d}_{\text{im}}(\boldsymbol{\theta}_n)]^T}_{\mathbf{A}} (\boldsymbol{\theta}_{n+1} - \boldsymbol{\theta}_n) \tag{16}$$

which can be solved as

$$\boldsymbol{\theta}_{n+1} = \boldsymbol{\theta}_n - (\mathbf{A}^T)^+ \mathbf{b}, \tag{17}$$

where $(.)^+$ is the pseudo-inverse. A near-exact null can then be found by obtaining an initial guess with the first-order solution, and then solving (17) iteratively. The method is easily extended to multiple nulls by including additional columns in $\mathbf{A}$ and elements in $\mathbf{b}$ in (16) for each additional direction $\hat{s}_k$, analogous to those for the first null.

IV. Illustrative Examples

This section provides some examples of the methods that were developed in the previous section.

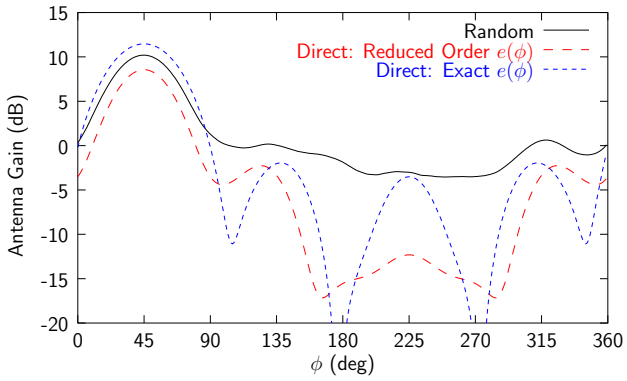

Fig. 4. Example application of geometrical solution showing synthesized beams for a steering angle of $\phi_0 = 45°$ for random search and the direct solution. For the direct solution, both the approximate reduced-order and exact radiation patterns are plotted.

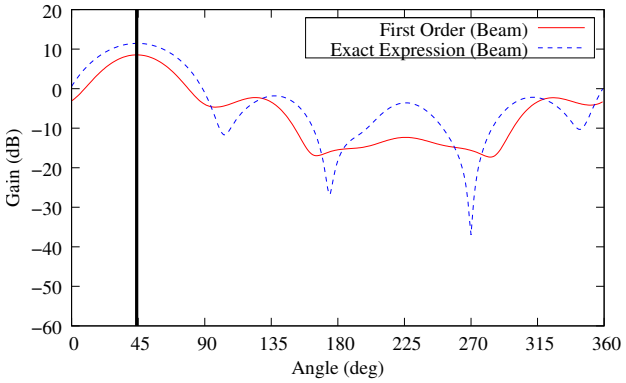

Fig. 5. Azimuthal gain pattern of the RECAP for the first-order and exact expression using the first-order beamforming solution for $\mathbf{\Gamma}$

TABLE I
PERFORMANCE OF DIRECT-NEWTON METHOD

N_{null}	1	2	4	8
G_{b}	8.7 dB	6.0 dB	-1.4 dB	-17.5 dB
$G_{\text{b-n}}$	88 dB	88 dB	87 dB	87 dB
M	3	4	7	45
t	39 ms	53 ms	85 ms	489 ms

A. Beamforming

As an example application of the method, we present beamforming for a rectangular parasitic dipole array shown in Figure 2. Figure 4 plots the gain of the parasitic antenna in the azimuthal plane for a steering angle of $\phi_{\text{b}} = 45°$. In this plot, "Random" denotes the gain of the best solution found with a random search of 1000 candidate RE configurations (results averaged over 100 trials). For the "Direct" solution curves, load reflection coefficients (in $\mathbf{\Gamma}$) are found using the simple geometric solution, where the curves for "Reduced-order" and "Exact" give the patterns computed for the same solution $\mathbf{\Gamma}$, but using (3) and (2), respectively, corresponding to the reduced-order pattern and the exact pattern.

The result shows that the direct method is able to find a better solution on average than the random search with 1000 candidates. Also, note that the run-times of the two algorithms with a MATLAB implementation are approximately 9.5 s and 12 ms for the random search with 1000 candidates and the direct method, respectively, indicating that the direct method is much more suitable for real-time implementation than unstructured search methods.

B. Joint Beamforming and Nullsteering

In this section, we consider joint beamforming and null steering with the 5×5 array, where the first-order approximation is used to find a main beam, which is subsequently refined with the Newton-based method to find one or more nulls.

Figure 5 shows the direct first-order solution for a main beam in direction $\phi = 45°$. As indicated in the plot, beamforming tends to be insensitive to small errors, and the first order solution provides a useful beamforming solution for the exact expression (2) as well. Next, from 1 to 8 nulls are formed by applying (17) iteratively until a minimum main-beam to null gain ($G_{\text{b-n}}$) separation of 60 dB is achieved. The method was run for 200 realizations consisting of a random main beam and random null positions having a minimum angular separation of $10°$, and average performance is given in Table I for different numbers of nulls. The result indicates that near exact nulls can be created with relatively few iterations (M) of the algorithm and low average run times per realization (t), but that for increasing numbers of nulls the gain in the specified main-beam direction (G_{b}) is significantly degraded.

The table suggests that when the amount of reconfigurability is high relative to the number of nulls, there are numerous roots of (2) available in search space and "excess reconfigurability" allows nulls to be found very efficiently with minimal impact on the first-order solution for the main beam. However, as the number of nulls increases and approaches the limits of the structure, the number of required iterations increases dramatically and the impact on the main beam is significant.

C. Comparison with a Genetic Algorithm

Genetic algorithms are well suited for difficult optimization problems with multiple goals such as RECAP optimization. Here, we compare the joint beamforming and nullsteering solution from the previous section with solutions obtained by a genetic algorithm (GA).

The GA here is similar to that employed in [7], and operates on a population of 100 candidate individuals, where the best 10 individuals are retained in each iteration (elitism). The chromosome vector consists of genes that are the real-valued phases $\boldsymbol{\theta}$, where the 5×5 array is scanned in a raster-like fashion. For each iteration, parents are selected randomly from the 10 best individuals, and the child chromosome is generated by copying element-by-element from one parent, where for each copy there is a probability of 0.2 of switching (crossing over) to the other parent, allowing multiple cross-overs to occur. Finally, each gene of the children is mutated with a probability of 0.2, and a mutation causes that element to assume a random phase uniformly distributed on $[0, 2\pi]$. The fitness function is the minimum gain separation from the main beam to nulls, given by

$$G_{\text{b-n}} = G_{\text{dB}}(\phi_{\text{beam}}) - \max_k G_{\text{dB}}(\phi_{\text{null},k}), \qquad (18)$$

TABLE II
COMPARISON OF GENETIC ALGORITHM AND DIRECT METHOD

N_{null}		1	2	4	8
G_{b}	GA	8.7 dB	5.9 dB	7.7 dB	9.9 dB
	Direct	11.4 dB	10.0 dB	9.1 dB	1.1 dB
$G_{\text{b-n}}$	GA	67 dB	60 dB	48 dB	23 dB
	Direct	96 dB	95 dB	73 dB	102 dB
t	GA	33 s	133 s	554 s	554 s
	Direct	0.033 s	0.049 s	0.084 s	0.336 s

where ϕ_{beam} is the direction of the desired main beam, $\phi_{\text{null},k}$ is the direction of the kth desired null, and the genetic algorithm is stopped when $G_{\text{b-n}}$ reaches 60 dB or 1000 iterations have been performed.

Figures 6(a)-(d) compare the direct solution with a single run of the genetic algorithm for a desired main beam at $45°$ and from 1 to 8 nulls at fixed angles $100°, 130°, \ldots, 310°$. The direct solution uses the same approach as in the previous section, where a main beam is found with the first-order solution and refined with the Newton-based method to find the nulls. From 1 to 4 nulls, there is remarkable similarity in the solutions found by the two distinct methods, with the direct method providing slightly higher gain and deeper nulls. For 8 nulls, the performance of the two methods diverges, since the GA is not able to find a suitable solution after 1000 iterations.

Table II gives the numerical performance of the genetic algorithm and direct method. As indicated, for relatively few nulls, the performance of the two methods is similar, except that the direct method is over 1000 times faster than the GA.

V. CONCLUSION

This paper has presented a direct method for optimization of reconfigurable parasitic antenna arrays using a reduced-order method that not only allows solutions for beamforming and nullsteering to be found very efficiently, but also provides valuable insight on the operation of the RECAP antenna. Numerical examples demonstrated that the proposed method strongly benefits from excess reconfigurability, providing many nulls in the search space that can be found rapidly with a Newton-based local optimizer. Comparison with a simple genetic algorithm indicated that the direct method can provide a factor of 1000 reduction in computational complexity. In future work, we will demonstrate how to extend the method to true in-situ cases where the optimizer does not have access to directional information like the antenna gain pattern or interferer directions, and must operate directly from SINR estimates. Future goals also include extension of the Newton-based method to constrain the main beam gain and incorporate practical limitations of the reconfigurable elements.

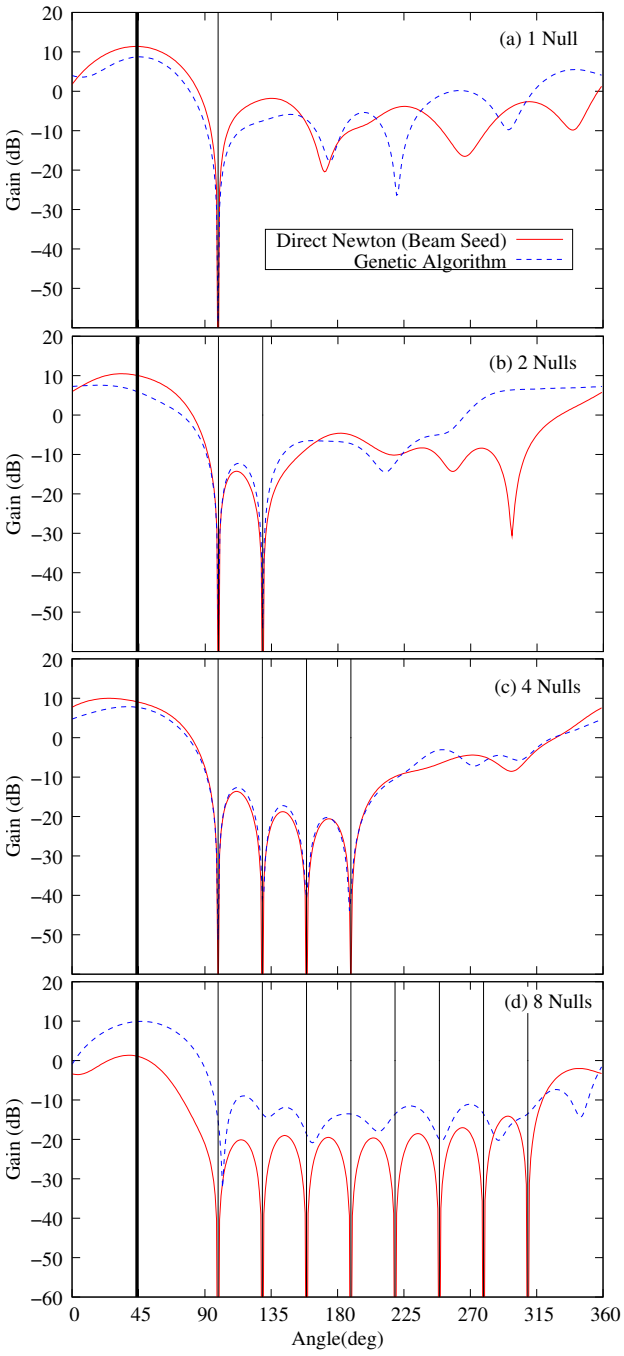

Fig. 6. Comparison of azimuthal gain patterns obtained with the direct method and a genetic algorithm for a fixed beam at $45°$ and varying numbers of nulls

REFERENCES

[1] R. Harrington, "Reactively controlled directive arrays," *IEEE Trans. Antennas Propag.*, vol. 26, pp. 390–395, May 1978.

[2] S. C. Panagiotou, T. D. Dimousios, S. A. Mitilineos, and C. N. Capsalis, "Broadband switched parasitic arrays for portable DVB-T receiver applications in the VHF/UHF bands," *IEEE Antennas and Propagation Magazine*, vol. 50, pp. 110–117, Oct. 2008.

[3] L. Petit, L. Dussopt, and J.-M. Laheurte, "MEMS-switched parasitic-antenna array for radiation pattern diversity," *IEEE Trans. Antennas Propag.*, vol. 52, pp. 2624–2631, Sept. 2006.

[4] R. Vaughan, "Switched parasitic elements for antenna diversity," *IEEE Trans. Antennas Propag.*, vol. 47, pp. 399–405, Feb. 1999.

[5] J. M. Johnson and Y. Rahmat-Samii, "Genetic algorithms in engineering electromagnetics," *IEEE Antennas and Propagation Magazine*, vol. 39, pp. 7–21, Aug. 1997.

[6] J. W. Wallace and R. Mehmood, "On the accuracy of equivalent circuit models for multi-antenna systems," *IEEE Trans. Antennas Propag.*, 2011, to appear.

[7] R. Mehmood and J. W. Wallace, "Diminishing returns with increasing complexity in reconfigurable aperture antennas," *IEEE Antennas Wireless Propag. Lett.*, pp. 299–302, 2010.