A Hybrid Global-Local Optimization Approach to Phase-Only Array-Pattern Synthesis

W. Mark Dorsey  
Naval Research Laboratory, Code 5317  
Washington, DC

Dan P. Scholnik  
Naval Research Laboratory, Code 5341  
Washington, DC

Abstract—Phased array magnitude/power-pattern synthesis is in general a nonconvex optimization problem which can exhibit a large number of local minima. This is often exacerbated by the addition of (nonconvex) magnitude constraints on the array weights. Practical large-scale design of such patterns requires both efficient local optimization to quickly identify candidate minima, and nonlocal (“global”) optimization to provide new starting points and evaluate the resulting minima to find a suitable, if not global, solution. Here we combine a very efficient constrained local formulation based on a weighted pattern-error metric with a parameterized heuristic method for generating initial array weights and a global method for searching over the parameter space. We additionally show that passband point-nulls, a common local-minima artifact, can be reliably detected early in the local optimization allowing for quick termination.

I. INTRODUCTION

Radar systems often utilize large phased arrays capable of supporting high-power transmission. The accompanying transmitters for these systems are typically run in saturation to maximize sensitivity and efficiency. As a result, the array only offers phase control of the element excitations. This complicates the synthesis of custom array patterns, particularly those with widened mainbeams, because all but the most trivial phase-only optimizations are non-convex and may have large numbers of local minima. Indeed, the existence of local minima problems is well documented in array synthesis [1], [2] and in related techniques such as phase retrieval [3], [4].

Global optimization techniques have been proposed to avoid the local minima problem [5]–[7], but high computational complexity often precludes their direct application to large designs. More promising are hybrid optimization techniques that combine efficient local solvers to quickly find local minima with heuristic and global methods for initializing the local optimizations and searching among the minima. These types of hybrid algorithms are applied in phaseless near-field measurements [8], [9], where phase-retrieval techniques are applied to measured near-field amplitude data of an antenna to recover the complex field distributions. For example, in [9], Razavi and Rahmat-Samii develop an analytical expression for the initial guess in terms of only a few input variables, and a global optimization finds the best initialization. The solution to the global technique is then used as the initial condition to the local solver. A similar technique is applied to reflectarrays in [10], where the coefficients of a Zernike polynomial initialization are optimized globally, and then used to initialize an alternating projection algorithm to design the phases of the elements.

In this work, we apply a hybrid global-local technique to the synthesis of a custom radiation pattern that covers a wide angular sector in azimuth, a cosecant-amplitude variation in elevation, and a deeply-suppressed null region at the horizon. The problem is initialized using functions derived from [11], which are defined in terms of a small number of variables thus making the application of a global solver at this level practical. The objective function for this global solver internally performs a fast local optimization for a fixed number of iterations, quickly identifying and discarding initializations that lead to the mainbeam nulls that are a common local-minimum artifact. Once the optimal initialization is found, it is then fed back into the local solver and run to full convergence. This paper will detail various parts of the algorithm and demonstrate its operation through an example calculation.

II. LOCAL OPTIMIZATION

The local optimization algorithm described in this section is an evolution of the one introduced in [12], itself descended from the nulling algorithm of [13], [14]. That formulation required significant hand-tuning of parameters to achieve convergence, largely due to a poor initialization choice. The modified algorithm described in this paper incorporates the more-general objective function from [15] along with a penalty-function approach to enforce the phase-only constraint. An optional deep sidelobe region constraint is enforced either by introduction of an additional penalty term or by restricting the step at each iteration to the subspace orthogonal to a linearization of the sidelobe constraints.

A. Array Factor

We assume in the sequel that we have identical isotropic elements in a planar array, so that the array pattern reduces to the 2D array factor

\[ A(\mathbf{u}) = \sum_{n=1}^{N} a_n e^{j\frac{2\pi}{\lambda} u^T d_n} \]  

(1)

where \( a_n \) is the complex array weight applied to the element at array-plane location \( d_n = (x_n, y_n) \). In the sequel a will represent the vector of all such weights. Here, \( \mathbf{u} = (u, v) \) represents two-dimensional spatial frequency normalized so that the unit
circle \(||u|| \leq 1\) corresponds to propagating radiation (so-called "visible" space). The array factor has the form of a discrete-space Fourier transform but with an inverse-transform exponent to describe plane-wave propagation.

### B. Constrained Optimization Formulation

The problem we wish to solve is

\[
\begin{align*}
\min_a & \quad f(a; W_o, p_o, q_o) \\
\text{s.t.} & \quad |a_n|^2 = \rho_n^2, \quad n = 1, \ldots, N \\
& \quad f(a; W_c, p_c, q_c) \leq \delta
\end{align*}
\]

(2a) (2b) (2c)

where \(f\) is the weighted pattern error

\[
f(a; W, p, q) = \left( \int W^p(u) \left| A(u) \right|^q - D^q(u) \right|^p \, du \right)^{\frac{1}{p}}.
\]

Here \(D\) is the desired magnitude pattern and \(W\) is a nonnegative weighting function. The parameters \(q\) and \(p\) permit some additional control over the nature of the resulting approximation, with \(q = \{1, 2\}\) choosing whether to approximate the magnitude or power pattern and \(p \geq 1\) controlling the relative importance of large and small errors. In particular, larger \(p\) emphasizes peak approximation errors (thus providing more of an equiripple error) while smaller \(p\) emphasizes the average errors. This weighted pattern-error metric is the same as that used in [15], with the exception that here it is a function of the complex array weights rather than of the phase of the weights. This in turn necessitates the constraints (2b) to enforce the fixed magnitude of the complex weights which were structurally imposed in [15]. The left side of constraint (2c) has the same form as the objective but with a different weighting function \(W\) and potentially different \(p\) and \(q\). The subscripts "o" and "c" on \((W, p, q)\) indicate objective and constraint, respectively.

Compared to the unconstrained optimization of [15], here the extra constraint (2c) allows the error in a specific region (a sidelobe region, say) to be explicitly bounded without tweaking a common weighting function. The degrees of freedom remaining after the constraints are met are applied to minimizing the error in another region (the mainlobe, say) resulting in a Pareto-efficient design. This is in contrast to alternating-projection approaches [16]–[18] which have no explicit objective and stop once the constraints are met, potentially giving up available performance.

A few steps are required to evolve the problem (2) into a form suitable for efficient solution. We outline these steps in the following sections.

### C. Discretization

Both the objective (2a) and constraint (2c) involve integrals of continuous-valued functions; for computational tractability we will reduce these to Reimann sums over a subset of an appropriately dense lattice. We can employ the FFT-based approach detailed in [15], which while restricting the element locations \(\{d_n\}\) to a lattice provides considerable computational savings when the weighting function \(W\) is nonzero on a significant number of points in the sum.

### D. Unconstrained Optimization Formulation

We wish to map our constrained problem to an equivalent unconstrained problem, so that we can solve it with conjugate-gradient or quasi-Newton methods. A straightforward approach is the so-called penalty method, in which the constraints are mapped to additional objective terms that penalize constraint violations [19], [20]. A common penalty function for equality constraints of the form \(g(a) = 0\) is the square function \(\frac{1}{2}g^2(a)\), which penalizes both positive and negative deviations. For inequality constraints of the form \(g(x) \leq 0\) the corresponding penalty function is

\[
\phi(g) = \frac{1}{2} \max\{0, g\}^2.
\]

It adds no penalty when the constraint is met, and grows as the constraint violation increases. Its derivative is the unit ramp function, which is also continuous:

\[
\frac{d\phi}{dg} = \max\{0, g\}.
\]

(4)

Rewriting the constraints (2b) and (2c) in standard (left-sided) form and applying this penalty-function transformation to (2) yields

\[
\begin{align*}
\min_a & \quad f_{\text{pen}} \triangleq f(a; W_o, p_o, q_o) + \\
& \quad \gamma \left[ \frac{1}{2} \sum_{n=1}^N (|a_n|^2 - \rho_n^2)^2 + \phi(f(a; W_c, p_c, q_c) - \delta) \right]
\end{align*}
\]

(5)

where \(\gamma\) is a positive scale factor that controls the relative importance of the objective and penalty terms. Initially this scale factor is set to a small value, and the unconstrained solution is obtained. Then \(\gamma\) is increased in a series of unconstrained minimizations, each initialized by the previous solution, to drive the constraint violations sufficiently low.

For quasi-Newton and conjugate-gradient solvers, we need also the gradient of this composite objective function. Each of the terms in (5) is a real function of the complex vector \(a\), and we will define corresponding complex gradients as detailed in [21] and as previously used in [14], [22]. The result is

\[
\nabla f_{\text{pen}} = \nabla f(a; W_o, p_o, q_o) + \gamma \left[ \frac{2}{2} \circ (|a|^2 - \rho^2) \right] + \max\{0, f(a; W_c, p_c, q_c) - \delta\} \nabla f(a; W_c, p_c, q_c)
\]

(6)

where \(\circ\) is the Hadamard (pointwise) vector product and the \(\cdot\) and \(\cdot^2\) operations are applied pointwise. The \(n\)th component of the weighted pattern-error gradient \(\nabla f\) can be written (suppressing the dependence on \(W, p,\) and \(q\) for space):

\[
\frac{\partial f(a)}{\partial a_n} = f^{1-p}(a) \int W^p(u) \left| A(u) \right|^q - D^q(u) \right|^{p-1} \times \text{sgn} \left( |A(u)|^q - D^q(u) \right) \frac{q A(u)}{|A(u)|^2} e^{-\frac{2 \pi}{\lambda} \sum c_n u_n} d\nu
\]

(7)

which has the form of an inverse Fourier transform back to the discrete spatial domain (again with an opposite sign in the exponential). We note that (5) involves a weighted sum of two gradients of the form (7), which can be computed quite
efficiently since a single FFT and IFFT can be shared between the two.

E. Alternate Approach for an Integrated Sidelobe Constraint

When the sidelobe constraint (2c) is defined such that $p_c q_c = 2$ and $D(u) = 0$ for any $u$ at which $W_c(u)$ is nonzero, then the constraint represents a bound on the (weighted) sidelobe energy, also known as the integrated sidelobe level. In discretized form, it can be written as $|\mathbf{H}_a|^2 \leq \delta$. As shown in [12], [14] such a constraint can be well approximated with a set of linear constraints of the form $\mathbf{V}_c^H a = 0$, where the $K < N$ columns of $\mathbf{V}_c$ are the right singular vectors of $\mathbf{H}$ corresponding to the $K$ largest singular values. The value of $K$ can be determined heuristically or empirically.

Linear equality constraints are particularly easy to enforce in conjugate-gradient and BFGS quasi-Newton methods. If the projection operation

$$\mathbf{P}_c a = (I - \mathbf{V}_c \mathbf{V}_c^H) a = a - \mathbf{V}_c \mathbf{V}_c^H a$$

is applied to the initial weights and to the gradient computed in each step, then both the search direction and the next point will lie in the subspace orthogonal to $\mathbf{V}_c$, thus enforcing the constraint. This can simplify the optimization vs. using a penalty term, but for large designs the $O(N^3)$ asymptotic complexity of the SVD tends to dominate the computation.

F. Local Solution Details

To locally solve the constrained optimization (2), we use the L-BFGS limited-memory quasi-Newton approach presented in [23] to solve a series of unconstrained subproblems of the form (5). The penalty weighting factor $\gamma$ is increased by a factor of 2–4 between stages, with each subproblem initialized by the previous result. The solution process stops when the constraint tolerances are met at the local minimum of a subproblem. A typical problem takes 5–8 stages to solve.

We use the L-BFGS routine from [24] since it directly supports the complex gradient notation of [21]. L-BFGS is very efficient for large problems, as each iteration (excluding the objective and gradient computation) has asymptotic complexity $O(N)$ vs. $O(N^2)$ for full-memory BFGS. Thus the overall iteration time is dominated by the $O(N \log N)$ objective and gradient.

III. GLOBAL OPTIMIZATION

The local optimization described in the previous section is quite efficient, but often returns results that are substantially suboptimal in the global sense. Thus additional mechanisms are required to ensure that an acceptable, if not globally optimal, solution is found. In this section we discuss the detection and mitigation of common point-null artifacts in local minima, introduce a heuristic method to initialize the local optimization, and consider the application of a global search algorithm over the space of initialization parameters.

A. Defective Local Minima: Passband Point-Nulls

Point nulls within the desired passband of the desired array pattern are a common characteristic of suboptimal local minima in phase-only pattern synthesis, phase retrieval, and similar problems. There are two aspects to dealing with these defective solutions: detection and mitigation.

1) Detection: The localized nature of the point null is such that it does not have a significant direct effect on the objective when $p_o$ is small. This makes it difficult to distinguish a solution with point-nulls to one without them by comparing only objective values. For large $p_o$ the nulls have a greater effect on the objective, albeit mitigated by their tendency to lie between sample points in spatial frequency.

To detect point nulls, we compute the ratio of the actual passband value ($A$) to the desired passband value ($D$). If the $\min(\frac{|A|}{D})$ within the passband is below a threshold, then we categorize the local minima as containing point-nulls. We then terminate the local solver early, without waiting for convergence. This approach is based on two empirical observations. First, that passband nulls form quickly, in a few iterations of the local solver. Second, that once formed, passband nulls almost always persist until convergence.

2) Mitigation: Several techniques have been proposed to mitigate the problem of passband point nulls. In [25] a multi-stage technique was proposed in which successively less-severe amplitude tapers are applied, effectively reducing the resolution of the array in the early stages. Each local optimization is seeded with the result of the previous until finally the full optimization is performed. A similar approach was suggested in [2] where the array size is gradually increased.

These suboptimal local minima are also encountered in phase-retrieval algorithms where they are seen as “stripes” in the image domain. Some potential mitigation techniques are outlined in [3], such as synthesizing a null-free spectrum from multiple spectra with nulls in different locations.

Another approach is to derive the initial vector $a_0$ from a model with many fewer degrees of freedom than array elements, so that it becomes practical to apply global optimization techniques to the parameters that generate the initial conditions while using local optimization to evaluate the resulting choices. In [10] Zernike polynomials of increasing order were used, with the polynomial coefficients optimized at each stage in a reflectarray design. In the next section we discuss a different parameterization that is more directly tuned to the desired response.

B. Initialization

The starting point used in [12] was linear phase, which turns out to be a poor choice for a phase-only pattern [15]. As a result, much manual tweaking of algorithm parameters was required across iterations to tease out a reasonable solution. Here we instead derive our initial phases for local minimization from a parameterized heuristic method that produces initial points that are closer to meeting the desired constraints. The major advantage of the parameterization is it significantly reduces the dimension of the space to be globally searched.
the examples of Sec. IV the reduction is from over a thousand
down to three.

A popular method for 1D signal synthesis simultaneously
in time and frequency is the method of stationary phase [26].
Although this approach requires impractically large arrays to
achieve high-quality patterns, it can provide reasonably close
starting points for local optimization. No general 2D extension
appears to be known, but in many cases we can use 1D
approximations separately for a planar array.

In [11], the stationary-phase method was used to derive
1D phase functions for phase-only synthesis of the principle-
plane radiation pattern from a circular phased array aperture of
radius $R$. That work focused on two specific one-dimensional
pattern shapes: a sector beam and a cosecant beam. For a
sector beam having uniform pattern amplitude for $-\pi R/2 \leq u \leq \pi R/2$, the phase function is obtained in closed-form as
$\psi_{sa}(x) = 2\pi R v_0 (-\frac{3}{8} x^2 + \frac{1}{16} x^4)$ for $x$ normalized to the
interval $[-1,1]$. To generate a pattern having a csc amplitude
taper for $v_0 \leq v \leq v_1$, the phase function is obtained from
numerical integration of

$$\psi'_a(y) = \frac{R}{\lambda} \eta'(y),$$

where

$$\eta'(y) = 2\pi \left[ -\frac{1}{v_0} + \frac{1}{v_0 - v_1} \int_{v_0}^{y} b^2(y) dy \right]^{-1},$$

Here $y$ is also normalized to $[-1,1]$ and $b(y) = 2\sqrt{1-y^2}$
is the effective 1D amplitude distribution that results from
collapsing the circular aperture onto the $y$-axis.

The passband shape desired for the example shown in the
following section has a uniform amplitude along the azimuth
dimension ($u$) and a csc amplitude taper in elevation ($v$).
Because of the circular array geometry, simply taking the
product $a_n = e^{j\psi_{sa}(u_n)} e^{j\psi_{sa}(v_n)}$ will not exactly yield a pattern
that is the outer product of the corresponding principle-plane
linear patterns. However, in practice it is close enough to be
useful. Such an initialization is shown in Fig. 1 for an aperture
of radius $R = 10\lambda$. Here, the initialization has been modified
to include a linear phase factor to scan the initialization off of
boresite and center it at $(u, v) = (u_s, v_s)$.

Only three free variables define the phase function: the
width of the sector beam is controlled by the value of $u_0$, while
the lower and upper edges of the csc-taper in elevation are
controlled by $v_0$ and $v_1$. The variables $u_s$ and $v_s$ from
the scan factor could be included as free variables in the overall
initialization phase function. However, in this work, they were
fixed at values that scanned the passband to the center of the
desired region.

C. Hybrid Global-Local Algorithm

The hybrid global-local algorithm begins with defining the
initialization in terms of $M \ll N$ variables to reduce the
computational complexity compared with globally optimizing
the entire excitation vector $a$. For the initialization defined
in the previous section, the excitation vector $a \triangleq a(p)$ is

$$a_n' = e^{j\psi_{sa}(u_n)} e^{j\psi_{sa}(v_n)}$$

a function of input-parameter vector $p = [v_0 v_1 u_0]$. Now
we globally optimize over the parameters $p$, with the global
objective being the the result of the local optimization (2)
when initialized with $a(p)$. The global optimization terminates
when a specified number of solutions have passed without
improvement after completing a threshold minimum number of
solutions. We can speed up the local optimization at this stage
both by early detection of mainbeam nulls and by restricting
the number of iterations or loosening convergence tolerances;
once the global solver selects the optimal parameter vector
$p_{opt}$ the local solver can be initialized with $a(p_{opt})$ and run to
full convergence.

Many global optimizers might be applied to this problem,
and indeed the authors experimented with several algorithms
implemented in the open-source NLopt toolbox [27] with simi-
lar results. For the examples of the next section the DIRECT
(Dividing RECTangles) algorithm [28] was chosen for the
global search. DIRECT systematically divides the bounded
search domain into progressively smaller hyperrectangles, with
some heuristics to determine the order of division. It guarante-
tees global convergence in a finite number of iterations under
various conditions on the objective.

The hybrid global-local approach outlined here can be
considered to be a member of the large class of multistart
methods [29], and is related to the basin-hopping algorithm
[30] in that the objective presented to the global optimizer is
a basin map.

IV. Example

As an example, we synthesize a pattern with a passband
that covers a sector between $0^\circ$ and $30^\circ$ in azimuth and $5^\circ$
and $60^\circ$ in elevation, with RMS sidelobes below $-20$ dB in
a band of elevation angles from $2^\circ$ below to $2^\circ$ above the
horizon. We use the linear-constraint approach of Sec. II-E to
enforce the sidelobe region. We seek a pattern that provides
uniform radar detection sensitivity at the lesser of 100 km and the
range at 40 km altitude. The result is a beam that is flat
for low elevation angles and falls off (as cosecant-squared in
magnitude) for higher elevation angles as the altitude limit
kicks in. The desired response $D$ is relaxed by 1 dB relative
to the ideal pattern to account for energy in the side lobes,
and we use parameters $p_0 = 2$ and $q_0 = 1$ to minimize the
RMS deviation from $D$. The array plane is inclined so that
boresight points $10^\circ$ above the horizon. The array elements
are centered on the 1075 points of the hexagonal lattice with
interelement spacing of $\lambda/\sqrt{3}$ that fall within a radius of
10A from the origin. The 1 dB and 3 dB beamwidths of the
array with uniform (identical) drive phases are about $1.7^\circ$ and
$3.0^\circ$, respectively. This example and array were previously
used in [15] where it was solved using an unconstrained local
objective and a much simpler initialization heuristic. Here, we
synthesize the desired pattern as a demonstration of the hybrid
global-local constrained algorithm.

We note that the initialization of Sec. III-B does not directly
account for the plane desired at low elevation angles. But
empirically it seems that the parameters $v_0$ and $v_1$ that
control the nominal start and stop elevation provide sufficient diversity. The plateau could be also accounted for simply by repeating Chakraborty’s procedure with the proper function, which would add a fourth parameter.

Prior to performing the optimization, a parametric study was performed to sample the solution space. This was done to determine appropriate settings for checking for passband point nulls. The results shown in Fig. 2 provide curves of $\min(\frac{|A|}{D})$ over the range of initialization parameters defined by $v_0 = 0.05$, $0.4 \leq v_1 \leq 1.0$, and $0.4 \leq u_0 \leq 1.0$. Here, we see a clear bifurcation of results, with the good and bad solutions showing clear separation early on in the local optimization. This largely validates the discrimination of point-null solutions after only a few iterations, at which point we can terminate the optimization early if point nulls are present. In four instances solutions did “escape” from the early formation of nulls after many iterations; further investigation showed that these examples had a single null close to the mainbeam edge, which eventually “drifted” into the sidelobes.

So a strategy based on early detection may miss some isolated potential solutions, although this appears to be an acceptable compromise.

The hybrid optimization algorithm was then used to determine the input-parameter vector $p_{\text{opt}}$ that minimizes the objective function while avoiding point nulls. The input-parameter vector $p$ was bounded between $p_{\text{lower}} = [0.00 \ 0.50 \ 0.50]$ and $p_{\text{upper}} = [0.10 \ 1.25 \ 1.25]$. Each local optimization call was limited to 125 iterations. Mainbeam nulls were detected using $\min(\frac{|A|}{D}) < -20 \text{ dB}$ after 60 iterations, and this test was repeated after the final iteration to catch any solutions with late-developing point nulls. Potential solutions failing the test were significantly penalized to ensure they are not selected by the global optimization. In addition to avoiding defective solutions, terminating the solution process early because of point nulls also saves time. Completing 125 iterations takes, on average, 12.1s on a 2.6 GHz Intel Core i7 laptop, while completion of only 60 iterations requires only 6.5s.

The global optimization required 125 iterations to determine $p_{\text{opt}} = [0.0981 \ 1.0139 \ 1.2083]$ to be the optimal starting point for synthesizing the desired pattern, and 15 of the iterations terminated early because of point-null detection. This set of input parameters—plotted in Fig. 1—was then used to initialize the local solver, which was run to full convergence (in this case, 664 iterations). Fig. 3(a) shows the pattern after 25 iterations. At this point, it’s clear that the passband is starting to take shape without the presence of point nulls, but more optimization is still required to clear up the sidelobe region and improve the passband (specifically at high elevation angles). The final pattern is shown in Fig. 3(b). The passband is still free of point nulls and the sidelobe region is significantly improved. In contrast, the patterns of Fig. 3(c) and Fig. 3(d) show results initialized by $p_{\text{bad}} = [0.0500 \ 0.5000 \ 0.5000]$ which results in point nulls within the passband. The point nulls are clearly apparent after 25 iterations which shows that these nulls form early in the optimization. Running this solution to full convergence improves the passband, but the point nulls remain virtually unchanged.

V. CONCLUSION

This paper presents a phase-only array-pattern synthesis technique that is a hybrid of an efficient local formulation with a parameterized heuristic initialization method and a global optimization method for searching the initialization-parameter space. The technique also successfully identifies and avoids defective local minima containing point nulls within the desired passband.
Fig. 3. Radiation patterns for two different input parameter vectors $p$. The patterns of (a)-(b) show results for the input parameters selected from the global optimization, and the patterns of (c)-(d) show results for a representative initialization resulting in passband point nulls.

ACKNOWLEDGMENT

This work was carried out under the Naval Research Laboratory Base Program sponsored by the Office of Naval Research.

REFERENCES