Antenna Array Output Power Minimization Using Steepest Descent Adaptive Algorithm

Sandra G. Johnson  A. David Snider

Department of Electrical Engineering
University of South Florida
Tampa, FL 33620 U.S.A.

Abstract—An adaptive algorithm to adjust the complex weights of an antenna array is presented that nulls high power signals while allowing reception of GPS signals as long as the signals arrive from different directions. The GPS signals are spread spectrum modulated and have very low average power, on the order of background thermal noise. Reducing the array output to this level will allow the GPS signals to be extracted.

The adaptive algorithm, named the Hilbert-space-based (HSB) gradient method, is based on the steepest descent algorithm and implements an efficient, exact gradient calculation [2]. The exact calculation of the gradient components, not approximations of them, distinguishes the HSB gradient algorithm from all other gradient based adaptive algorithms. Furthermore, the algorithm calculates the gradient components simultaneously and efficiently.

The HSB gradient algorithm is implemented in Matlab and output plots illustrating the performance of the algorithm are included.

I. INTRODUCTION

An antenna is a device to transmit and receive electromagnetic radiation. The antenna creates a pattern of radiation that spatially describes areas of gain (lobes) or attenuation (nulls) for the signals it transmits or receives. The radiation pattern of an array of multiple antennas consists of a weighted superposition of each antenna’s radiation pattern. The location of lobes and nulls in an array radiation pattern can be controlled with a beamforming network consisting of a signal processor implementing an algorithm that weights the outputs of each antenna and combines them to form the array output. This paper discussed the signal processing algorithm; the RF data received by the antennas is assumed to be downconverted to a digital baseband stream and the signal processing constituting the proposed algorithm occurs completely in the digital domain.

The objective of the Hilbert-space-based (HSB) gradient adaptive algorithm is to receive signals from the Global Positioning System (GPS) of satellites by nulling all jammer signals received by the array. The desired GPS signals have very low power, on the order of thermal noise, -160 dBw, so if the output power of the array is above this level, it is assumed to come from jammer signals. The adaptive algorithm reduces the power output of the array by adapting weights of the antennas to null power. The adaptive algorithm is not based on forming lobes in the direction of arrival of the GPS signals.

II. ANTENNA ARRAY FUNDAMENTALS

A beamforming network assigns an adjustable weight to each antenna in an array. Each weight is a complex number which has a variable gain (magnitude) and a variable phase. The superposition of the individual antenna outputs scaled by the adjustable weights forms the array output. Adjusting the weight of each antenna has the effect of steering lobes and nulls of the array pattern to various positions. Signals arriving at the array in a null location will be severely attenuated or even eliminated by the array; signals arriving at a lobe will be enhanced by the array. Thus the antenna array performs spatial filtering using the mainlobe, sidelobes, and nulls of its radiation pattern. The spatial areas in the array pattern of gain and attenuation are frequency dependent. A deep null at \( \omega_1 \) may not be as deep, or even a null, at another frequency \( \omega_2 \). The bandwidths of the desired and jammer signals must be taken into account when designing the antenna array as well as the beamforming network. Each antenna weight in a narrowband array is a complex number; the antenna weights controlled by the proposed adaptive algorithm are implemented as such in the simulations to follow. For wideband arrays, the antenna weights must provide the ability to null jammers over a range of frequencies. These weights, being frequency dependent, can be implemented at each antenna as a linear filter or a digital tapped delay line.

The choice of antenna weights can significantly affect the array output. Antenna phases can be adjusted to compensate for signal propagation delays between antennas so that the signal appears amplified at the output of the array. The antenna weights can also be adjusted so that the antenna outputs cancel each other eliminating the array output. Normally, it is up to the weight adjustment algorithm to distinguish friendly signals from jammer signals for proper lobe and null placement. The proposed algorithm does not distinguish the type of incoming signal although it assumes different power levels between the desired and unwanted signals; it simply tracks the output power of the array and adjusts the antenna weights until
the output power is minimized.

The antennas (assumed identical) in an array can be arranged in any configuration. For the HSB algorithm, one antenna in the array is selected as the reference antenna. The other antennas in the array are referred to as peripheral antennas. All peripheral antennas have adjustable gain $w_m$ and adjustable phase $q_m$ represented as a complex term $A_m = w_me^{iq_m}$, $m = 1, 2, \ldots, N_{ant}$ where $N_{ant}$ is the total number of antennas in the array. The weight of the reference antenna, $m = 1$, is fixed at a constant value, $A_1 \equiv 1$.

The input vector to the array, $x$, consists of GPS signals, noise, and jammers. The output of the array is $xA$, where $A$ represents the vector of antenna weights, $[A_1 \ A_2 \cdots \ A_{N_{ant}}]$.

Since the array output is dominated by the high power jammers, the task of the algorithm is to adjust the antenna weights, $A_2, \ldots, A_{N_{ant}}$, so that nulls are steered toward the jammer arrival directions, minimizing the jammer power to the level of the noise and GPS signals. This leaves the GPS signals intact to be processed by spread spectrum demodulation techniques. Recall the reference antenna weight is fixed at $A_1 = 1$, so that the algorithm minimizes the output power while avoiding the trivial solution, $A_m = 0$, $m = 1, 2, \ldots, N_{ant}$.

### III. HSB ALGORITHM

The power minimization and weight calculations of the HSB gradient algorithm will proceed as follows. For some initial choice of weights $A$, the output power of the array is sampled every $\tau$ seconds and assessed at times labeled by the variable $t_\gamma = \gamma \tau$ ($\gamma$ is an integer), $\Gamma$ of these power measurements are averaged every $\Gamma \tau$ seconds, and the gradient of this average (with respect to the weights) is calculated. The peripheral antenna weights are adjusted in the direction of the negative gradient. This constitutes one algorithm iteration. It is unlikely that one single adjustment in each weight value results in their optimal values so these updated weights are used to accumulate another batch of $\Gamma$ output power measurements, and the algorithm is repeated. If the procedure converges, the search becomes more precise, with finer adjustments in the weight values toward their optimal values, as the algorithm progresses.

The instantaneous output power of the array, $p(t_\gamma, A)$, is

$$ p(t_\gamma, A) = (xA)(xA)^\dagger $$

where $^\dagger$ represents conjugate transpose. For each value of $t_\gamma$, $p(t_\gamma, A)$ represents one sample of the instantaneous array output power.

During the $k$th iteration of the algorithm, $\Gamma$ samples of instantaneous power, $p(t_\gamma, A)$, are averaged to obtain the average output power of the array, $P(k, A)$, i.e.,

$$ P(k, A) = \frac{1}{\Gamma} \sum_{t_\gamma = (k-1)\Gamma+1}^{k\Gamma} p(t_\gamma, A). $$

The average output power is a quadratic function of the antenna weights. The strategy is to converge to the minimum by continuously adjusting the values for the antenna weights so as to enforce $P(k + 1, A) < P(k, A)$. 

Recall that $A$ is a $1 \times N_{ant}$ vector with components $w_me^{iq_m}$, $m = 1, 2, \ldots, N_{ant}$. The power $P(k, A)$ can be expressed explicitly in terms of the magnitudes, $w_m$, and phases, $q_m$, of the antenna weights by reincarnating $A_m = w_me^{iq_m}$ as a new variable $W$, where $W$ is a column of antenna weight parameters $[w_1 \ w_2 \cdots w_{N_{ant}} \ q_1 \ q_2 \cdots q_{N_{ant}}]^T$ (recall since $A_1 \equiv 1$, $w_1 = 1$ and $q_1 = 0$). The objective of the algorithm is to choose the change in the weights, $\Delta W_m$, to reduce $P(k, W_m)$:

$$ P(k + 1, W_m + \Delta W_m) \leq P(k, W_m), \quad m = 1, 2, \ldots, 2N_{ant} $$

Without risk of confusion, we write $P(W)$ for $P(k, A)$. Expanding $P(W_m + \Delta W_m)$ with a (matrix) Taylor series to first order yields

$$ P(W_m + \Delta W_m) \approx P(W_m) + \Delta W^T \frac{\partial P}{\partial W} $$

where $W$ is the column of weight parameters introduced above and $\nabla P$ represents the column of all components of the gradient of the output power with respect to the antenna weight parameters, $\nabla P = [\frac{\partial P}{\partial w_1}; \frac{\partial P}{\partial w_2}; \cdots; \frac{\partial P}{\partial w_{N_{ant}}}; \frac{\partial P}{\partial q_1}; \frac{\partial P}{\partial q_2}; \cdots; \frac{\partial P}{\partial q_{N_{ant}}}]^T$.

For the steepest descent algorithm, the weights from iteration $k$ to iteration $k + 1$ are adjusted in the direction of the negative power gradient with a scale factor $\mu$, the “step size”

$$ \Delta W = -\mu \nabla P. $$

Substituting equation (4) for $\Delta W^T$ in equation (3) results in

$$ \Delta P = P(W + \Delta W) - P(W) \approx -\mu \nabla P^T \nabla P. $$

The desired change in power is known; it is the difference between the present power, $P(k, A) = P(W)$ and the minimum power, which is approximately zero.

$$ \Delta P = \text{change in power} = (0 - P(W)) \approx -\mu \nabla P^T \nabla P $$

and solving for the step size $\mu$, one obtains

$$ \mu = \frac{P(W)}{\nabla P^T \nabla P}. $$

The weight adjustment equation (4) becomes

$$ \Delta W = -\mu \nabla P = -\frac{P(W)}{\nabla P^T \nabla P} \nabla P. $$

Equation (7) in terms of the weight magnitudes states

$$ \Delta w_i = -\mu \frac{\partial P}{\partial w_i} = -P(W) \times \left( \sum_{m=2}^{N_{ant}} \frac{\partial P/\partial w_i}{[\partial P/\partial w_m]^2 + [\partial P/\partial q_m]^2} \right) $$

and in terms of the weight phases it states

$$ \Delta q_i = -\mu \frac{\partial P}{\partial q_i} = -P(W) \times \left( \sum_{m=2}^{N_{ant}} \frac{\partial P/\partial q_i}{[\partial P/\partial w_m]^2 + [\partial P/\partial q_m]^2} \right) $$
occurs once per algorithm iteration, each time with updated values for $\mu$ and $\nabla P$. Our innovation for calculating the gradient components, $\frac{\partial P}{\partial w_m}$ and $\frac{\partial P}{\partial q_m}$, will be described in the next section.

If the array is in motion, the optimal weight search performed by the algorithm becomes nonstationary because the power surface changes at each iteration. For an array in motion, the power is time averaged and each weight is updated per algorithm iteration as it is for a stationary array.

IV. EXACT GRADIENT CALCULATION

The gradient of the average array output power, $\nabla P$, is computed using the Hilbert space inner products of signals which are readily available at the array output by the following method [1]. From equations (1) and (2),

$$P(k, A) = \frac{1}{T} \sum_{t=(k-1)T+1}^{kT} p(t, A) = \frac{1}{T} \sum_{t=(k-1)T+1}^{kT} (xA)(xA)$$

Rewrite this summation using inner product bra/ket notation

$$<a, b> = \frac{1}{T} \sum_{t=(k-1)T+1}^{kT} a(t)b^\dagger(t)$$

and the power is represented as

$$P(k, A) = <xA, xA>.$$

Since the array output results from the $N_{ant}$ antenna outputs,

$$xA = \sum_{m=1}^{N_{ant}} A_m x_m(k),$$

the array output power becomes

$$P(k, A) = \sum_{g=1}^{N_{ant}} A_g x_g(k) \sum_{g=1}^{N_{ant}} A_g x_g(k)$$

$$= \sum_{g=1}^{N_{ant}} A_g A_g^\dagger <x_g(k), x_g(k)>.$$

Now the $m$th component of the gradient with respect to the weight magnitude $w_m$ is (the time dependence of $x$ is dropped for clarity) (recall $A_g = w_g e^{iq_g}$)

$$\frac{\partial P}{\partial w_m} = \frac{\partial}{\partial w_m} \left( \sum_{g=1}^{N_{ant}} \sum_{h=1}^{N_{ant}} A_g A_h^\dagger <x_g, x_h> \right)$$

$$= \sum_{g} \sum_{h} <x_g, x_h> A_g \frac{\partial w_h e^{-iq_h}}{\partial w_m}$$

$$+ \sum_{g} \sum_{h} <x_g, x_h> \frac{\partial w_g e^{iq_g}}{\partial w_m} A_h^\dagger$$

$$= \sum_{g} <x_g, x_m> A_g e^{-iq_m}$$

$$+ \sum_{h} <x_m, x_h> e^{iq_m} A_h^\dagger.$$

Rearranging terms and identifying the summation of the antenna outputs as the array output signal, $xA$ from equation (10), yields

$$\frac{\partial P}{\partial w_m} = \sum_{g} <x_g, x_m> + \sum_{h} <e^{iq_m} A_h x_h>$$

$$= <xA, A_m x_m> / w_m + <A_m x_m, xA> / w_m.$$

The components of the gradient with respect to the weight magnitudes are simply the Hilbert space inner products of the real parts of the appropriately scaled (by $1/w_m$) individual output signals of each antenna $A_m x_m(k)$ with the array signal output, $xA(k)$. That is, they are correlations (zero delay).

Similarly, the $m$th component of the gradient with respect to the phase $q_m$ of the weight is

$$\frac{\partial P}{\partial q_m} = \frac{\partial}{\partial q_m} \left( \sum_{g=1}^{N_{ant}} \sum_{h=1}^{N_{ant}} A_g A_h^\dagger <x_g, x_h> \right)$$

$$= \sum_{g} \sum_{h} <x_g, x_h> A_g \frac{\partial w_h e^{-iq_h}}{\partial q_m}$$

$$+ \sum_{g} \sum_{h} <x_g, x_h> \frac{\partial w_g e^{iq_g}}{\partial q_m} A_h^\dagger$$

$$= -i \sum_{g} <x_g, x_m> A_g A_g^\dagger$$

$$+ i \sum_{h} <x_m, x_h> A_h A_h^\dagger.$$

Rearranging terms and identifying the summation of the antenna outputs as the array output signal, $xA$ from equation (10), yields

$$\frac{\partial P}{\partial q_m} = -i \sum_{g} <A_g x_g, A_m x_m> + i \sum_{h} <A_m x_m, A_h x_h>$$

$$= -i <xA, A_m x_m> + i <A_m x_m, xA>$$

$$= 2i <A_m x_m, xA>.$$

The components of the gradient with respect to the weight phases are simply the Hilbert space inner products of the imaginary parts of the scaled individual output signals of each antenna $A_m x_m(k)$ with the array signal output, $xA(k)$.

With all components of the gradient available, they are then used to determine the antenna weights to minimize the output power from the array as in equations (8) and (9).

V. HSB ALGORITHM PERFORMANCE

Three plots of the output of the algorithm simulated in Matlab are included in this section. The array used in
these simulations consisted of 7 antennas in a planar configuration of 1 central, fixed-weight antenna and 6 peripheral antennas equally spaced around the central antenna at $0^\circ$, $60^\circ$, $120^\circ$, $180^\circ$, $240^\circ$, and $300^\circ$. The jammers arrive in the plane of the array with azimuth angles $\phi = 120^\circ$, $330^\circ$, $25^\circ$, $204^\circ$, $0^\circ$, and $108^\circ$. The jammer amplitudes are simulated with constant amplitudes of 100 V (or 40 dB). The noise added to the input of the array is 1.25 (or 1 dB) and the algorithm attempts to reduce the output power level to this level. The number of iterations simulated is 25000, a duration of 0.5 seconds. The motion of the array is indicated in the plot legends. As the plots show, the algorithm successfully reduces the output power of the array to the noise floor.

![Figure 1: Six Jammers, No Motion.](image1)

![Figure 2: Six Jammers, Yaw Motion.](image2)

**VI. CONCLUSION**

The implementation of the HSB algorithm in Matlab resulted in the reduction of the array output power to the noise floor thus achieving the desired output.

**References**
