AN ESCALATOR STRUCTURE FOR ADAPTIVE BEAMFORMING

B.B. Madan
Department of Electrical Engineering, Indian Institute of Technology, Delhi, New Delhi-110016, India

G.V. Kuriyan
Centre for Applied Research in Electronics, Indian Institute of Technology, New Delhi-110016, India

ABSTRACT

The standard IIIS algorithms for adaptive beamforming though straightforward, suffer from the main drawback of having slow convergence rate. This is normally attributed to the wide range of spread of eigenvalues of the input signal vector correlation matrix. By properly pre-whitening the input signal vector these eigenvalues can be equalized thereby improving the convergence rate. An Escalator structure recently proposed for linear prediction problems is used to pre-whiten the input signal vector. This whitened signal vector is now fed to a beam former with desired main beam shape. The overall adaptive array is computationally efficient and exhibits faster convergence rate than the conventional INS algorithms.

INTRODUCTION

Adaptive arrays have the desirable property of responding to the operating environment by automatically placing nulls in the direction of undesirable interfering noise sources without affecting the desired signal. The response of an adaptive array can be controlled by introducing complex weights at the outputs of different elements of an array. Fig.1 shows shows typical n-element adaptive array.

The input signal vector \(X(k)=[X(k,1), X(k,2), ..., X(k,n)]\) at the kth sampling instant consists of the desired signal, interfering noise sources and ambient broad-band noise. The weight vector \(W'(k)=[W(k,1), W(k,2), ..., W(k,n)]\) is to be computed such that \(E\{y(k) - d(k)\}^2\) is minimized. Here, \(y(k)\) is the actual output and \(d(k)\) is the output desired. The optimum height vector \(W_{opt}\) is given by

\[
W_{opt} = R_{xd}^{-1} R_{xx}
\]

where, \(R_{xx} = E\{X(k) X'(k)\}\) is an (nxn) input signal correlation matrix and \(R_{xd} = E\{X(k), d(k)\}\) is an (nxl) cross correlation matrix.

To compute the weight vector \(W(k)\) directly, one will normally have to invert an (nxn) matrix. To avoid this inversion, an estimate \(\hat{W}(k)\) of the optimum weight vector \(W_{opt}\) can be obtained by using well known graaient descent algorithm [ii. This algorithm, iteratively updates the weights vector \(W(k)\) as,

\[
W(k+1) = W(k) + uX(k)(y(k) - d(k))
\]

where \(u\) is referred to as the step size and it controls the convergence rate and the self, noise of the algorithm.

In the algorithm the knowledge of \(d(k)\) is essential which however may not be available explicitly, we can instead specify that the desired signal is arriving from noise known direction, which for the sake of convenience can be assumed to be the array broadside. In that case normalized \(d(k)\) can be indirectly specified as a direction vector [2]

\[
d(k) = [1 1 1 ... 1]
\]

There are other techniques available e.g. constrained iNS algorithms [3] which can be made use of in case \(d(k)\) is not available explicitly.

Most of these algorithms use gradient descent techniques and have the drawback of having very slow convergence rate towards the optimal solution. This is more so for weak noise sources, which may take a long time to be nulled out. In fact it has
been shown that the convergence rate behaviour is controlled by the range of spread of the eigenvalues of $R$. Various techniques have been suggested for de-spreading the range of eigenvalues in order to increase the convergence rate. All the eigenvalues can be made equal if we design a pre-whitening filter which whitens the input signal vector $X(k)$. This can be followed by an appropriate beamformer which has desired main beam response.

II. ESCALATOR STRUCTURE FOR ADAPTIVE ARRAY

In the last section, a similarity between the linear prediction and adaptive beam-forming problems has been highlighted. Recently Ahmed and Youn [6] have proposed a new realization for linear prediction which they refer to as the Adaptive Escalator Predictor (AEP). An N-element adaptive array based on this escalator realization is shown in Fig.2. An important property of the AEP realizations is that the error outputs $y_1(k), y_2(k), \ldots, y_{N-1}(k)$ are mutually uncorrelated. In other words, for the input signal vector $X(k) = [x_1(k), x_2(k), \ldots, x_N(k)]$ which is assumed to be a zero mean random process, the output vector $y(k) = [y_1(k), y_2(k), \ldots, y_{N-1}(k)]$ given by

$$y(k) = w \cdot X(k)$$

such that

$$R_{yy} = w^* W^*$$

is a diagonal matrix, where

$$R_{yy} = E(y(k) y^T(k))$$
$$R_{xx} = E(x(k) x^T(k))$$

In (a), $W$ represents the complex conjugate transpose of the transformation matrix $W$.

For an N-element array, the escalation predictor has $(N-1)$ stages. At every stage $i=1,2,\ldots,N-1$, we need to compute $(N-i)$ complex weights $w_{ij}$, $j=1,2,\ldots,N-i$. These weights are computed by solving a set of scalar prediction problems. For example, at the 1st stage, the weights $w_{1j}$ are computed by solving,

$$E\{[X_i(k) - w_{1j} x_{1j}(k)][X_{N-1}(k) - w_{1j} x_{N-1}(k)]\}$$

where superscript denotes complex conjugate operation. On performing the indicated minimization with respect to both real and imaginary part of the complex weights $w_{ij}$, we get the following set of equations which are used to iteratively update the weights $f6$

$$w_{ij}(k+1) = w_{ij}(k) + \mu [y_{ij}(k)y_{i-1,j}(k)]$$

where $y_{ij}(k)$ is the prediction error given by

$$y_{ij}(k) = y_{i-1,j}(k) - y_{i,j}(k)y_{i-1,i}(k)$$

and

$$y_{ij}(k) = x_{i,j}(k)$$

To ensure convergence, we can normalize the step size $\mu$ [7], so that

$$w_{ij}(k+1) = w_{ij}(k) + \frac{\mu}{V}(k)[y_{ij}(k)y_{i-1,j}(k)]$$

where $V(k)$ is to denote the variance of the prediction error $y_{ij}(k)$, which can be estimated from the illimiring data as follows:

$$V(k+l)=V(n)+V(k)$$

all mutually uncorrelated. In other words, the vector $Y(k)$ is the whitened version of $X(k)$. Consequently, $X^*$ has all its eigen-values equal.

The complete adaptive array consists of this whitening preprocessor cascaded to an appropriate beamformer. This beamformer can be designed using uniform, Chebychev or some other weights so that it has the desired main beam response.

III. SIMULATION

The proposed approach to adaptive beam forming was tested for a 4-element array. The simulated environment consisted of a broadside desired signal and two interfering noise sources. Finally, white noise was added to simulate the ambient and other broad noise sources. This simulated environment is summarized in Table I. The overall response of this array for $\mu=0.02$ and $\nu=0.98$ is shown in Fig.3. Within about 10 iterations, nulls in the direction of interfering noise sources begin to become apparent and deep nulls are formed within about 20 iterations.

<table>
<thead>
<tr>
<th>Source</th>
<th>Power</th>
<th>Direction</th>
</tr>
</thead>
<tbody>
<tr>
<td>Broad Side Signal</td>
<td>0.1</td>
<td>0°</td>
</tr>
<tr>
<td>Noise Source A</td>
<td>5.0</td>
<td>-30°</td>
</tr>
<tr>
<td>Noise Source B</td>
<td>3.0</td>
<td>+90°</td>
</tr>
<tr>
<td>White Noise (every element)</td>
<td>0.1</td>
<td>-</td>
</tr>
</tbody>
</table>

IV. CONCLUSION

A new technique for adaptive beamforming based on complex version of the AEP realization has been described. The proposed technique rapidly adapts itself
to operating environment. This technique is computationally efficient because the weights are computed by a set of decoupled equations obtained by solving only scalar linear prediction problems. This point also suggests that weights may also be computed by working directly with the estimated correlation matrices (actually coefficients for this realization). This modified approach is likely to be faster and robust, the later so because we need no longer worry about the step size $U$.

REFERENCES


---

Fig. 2 - 4-element adaptive array based on AEP realization.

---

Fig. 3 - Adaptive array response.

8.8