CHAPTER 5

ADAPTIVE BEAMFORMING

Adaptive Beamforming is a technique in which an array of antennas is exploited to achieve maximum reception in a specified direction by estimating the signal arrival from a desired direction (in the presence of noise) while signals of the same frequency from other directions are rejected. This is achieved by varying the weights of each of the sensors (antennas) used in the array. It basically uses the idea that, though the signals emanating from different transmitters occupy the same frequency channel, they still arrive from different directions. This spatial separation is exploited to separate the desired signal from the interfering signals. In adaptive beamforming the optimum weights are iteratively computed using complex algorithms based upon different criteria.

Beamforming is generally accomplished by phasing the feed to each element of an array so that signals received or transmitted from all elements will be in phase in a particular direction. The phases (the interelement phase) and usually amplitudes are adjusted to optimize the received signal. The array factor for an *N*-element equally spaced linear array is given,

$$AF(\phi) = \sum_{n=0}^{N-1} A_n e^{jn(\frac{2\pi d}{\lambda}\cos\phi + \alpha)} \tag{5.1}$$

Note that variable amplitude excitation is used.

The interelement phase shift is given by,

$$\alpha = -\frac{2\pi d}{\lambda_0}\cos\phi_0 \tag{5.2}$$

ϕ_0 is the desired beam direction. At wavelength λ_0 the phase shift corresponds to a time delay that will steer the beam to ϕ_0.

5.1 Adaptive beamforming problem setup

To illustrate different beamforming aspects, let us consider an adaptive beamforming configuration shown below in figure 5.1.

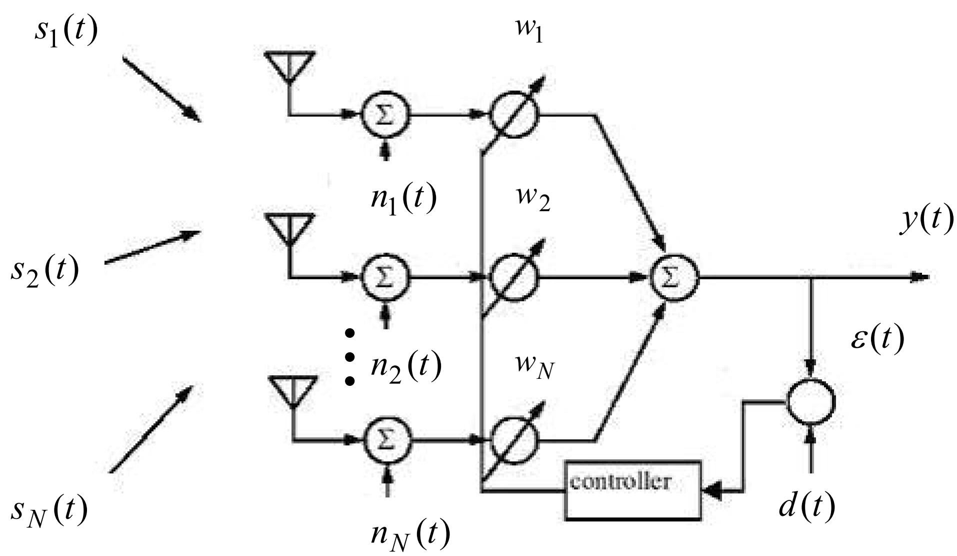

Figure 5.1 An Adaptive array system

The output of the array $y(t)$ with variable element weights is the weighted sum of the received signals $s_i(t)$ at the array elements and the noise $n(t)$ at the receivers connected to each element. The weights w_m are iteratively computed based on the array output $y(t)$, a reference signal $d(t)$ that approximates the desired signal, and previous weights. The reference signal is approximated to the desired signal using a training sequence or a spreading code, which is known at the receiver. The format of the reference signal varies and depends upon the system where adaptive beamforming is implemented. The reference signal usually has a good correlation with the desired signal and the degree of correlation influences the accuracy and the convergence of the algorithm.

The array output is given by

$$y(t) = w^H x(t) \tag{5.3}$$

Where w^H denotes the complex conjugate transpose of the weight vector w.

In order to compute the optimum weights, the array response vector from the sampled data of the array output has to be known. The array response vector is a function of the incident angle as well as the frequency. The baseband received signal at the N-th antenna is a sum of phase-shifted and attenuated versions of the original signal $s_i(t)$.

$$x_N(t) \cong \sum_{i=1}^{N} a_N(\theta_i) s_i(t) e^{-j2\pi f_c \tau_N(\theta_i)} \tag{5.4}$$

The $s_i(t)$ consists of both the desired and the interfering signals.

$\tau_k(\theta_i)$ is the delay, f_c is the carrier frequency.

$$a(\theta_i) = [a_1(\theta_i)\ e^{-j2\pi f_c \tau_1(\theta_i)}, a_2(\theta_i)\ e^{-j2\pi f_c \tau_2(\theta_i)} \ldots\ldots\ldots\ldots\ldots a_N(\theta_i)\ e^{-j2\pi f_c \tau_N(\theta_i)}]^T \tag{5.5}$$

Now,

$$A(\theta) = [a(\theta_1) a(\theta_2) \ldots. a(\theta_d)] \tag{5.6}$$

$$S(t) = [s_1(t) s_2(t) \ldots. s_d(t)]^T \tag{5.7}$$

So that,

$$x(t) = A(\theta) S(t) \tag{5.8}$$

With noise,

$$x(t) = A(\theta)S(t) + n(t) \tag{5.9}$$

$a(\theta)$ is referred to as the array propagation vector or the steering vector for a particular value of θ.

The beamformer response can be expressed in the vector form as,

$$r(\theta,\omega) = w^H a(\theta,\omega) \tag{5.10}$$

This includes the possible dependency of $a(\theta)$ on ω as well.

To have a better understanding let us re-write $x(t)$ in equation 5.9 by separating the desired signal from the interfering signals. Let $s(t)$ denote the desired signal arriving at an angle of incidence θ_0 at the array and the $u_i(t)$ denotes the N_u number of undesired interfering signals arriving at angles of incidence θ_i. It must be noted that, in this case, the directions of arrival are known a priori using a direction of arrival (DOA) algorithm.

The output of the antenna array $x(t)$ can now be re-written as;

$$x(t) = s(t)a(\theta_0) + \sum_{i=1}^{N_u} u_i(t)a(\theta_i) + n(t) \tag{5.11}$$

where,

$a(\theta_i)$ is the array propagation vector of the i^{th} interfering signal.

$a(\theta_0)$ is the array propagation vector of the desired signal.

Therefore, having the above information, adaptive algorithms are required to estimate $s(t)$ from $x(t)$ while minimizing the error between the estimate $\hat{s}(t)$ and the original signal $s(t)$.

Let $d^*(t)$ represent a signal that is closely correlated to the original desired signal $s(t)$. $d^*(t)$ is referred to as the reference signal, the mean square error (MSE) $\varepsilon^2(t)$ between the beamformer output and the reference signal can now be computed as follows;

$$\varepsilon^2(t) = [d^*(t) - w^H x(t)]^2 \tag{5.12}$$

After taking an expectation on both sides of the equation we get,

$$E\{\varepsilon^2(t)\} = E\{[d^*(t) - w^H x(t)]^2\} \tag{5.13}$$

$$E\{\varepsilon^2(t)\} = E\{[d^2(t)\} - 2w^H r + w^H Rw \tag{5.14}$$

where $r = E\{[d*(t)x(t)]\}$ is the cross-correlation matrix between the desired signal and the received signal and $R = E[x(t)x^H(t)]$ is the auto-correlation matrix of the received signal also known as the covariance matrix. The minimum MSE can be obtained by setting the gradient vector of the above equation with respect to w equal to zero, i.e.

$$\nabla_w(E\{\varepsilon^*(t)\}) = -2r + 2Rw \tag{5.15}$$

$$= 0$$

Therefore the optimum solution for the weight w_{opt} is given by

$$w_{opt} = R^{-1}r \tag{5.16}$$

This equation is referred to as the optimum Weiner solution.

5.2 Traditional adaptive beamforming approaches

The following discussion explains various beamforming approaches and adaptive algorithms in a brief manner.

5.2.1 Side lobe cancellers

This simple beamformer shown below consists of a main antenna and one or more auxiliary antennas. The main antenna is highly directional and is pointed in the desired signal direction. It is assumed that the main antenna receives both the desired signal and the interfering signals through its sidelobes. The auxiliary antenna primarily receives the interfering signals since it has very low gain in the direction of the desired signal. The auxiliary array weights are chosen such that they cancel the interfering signals that are present in the sidelobes of the main array response.

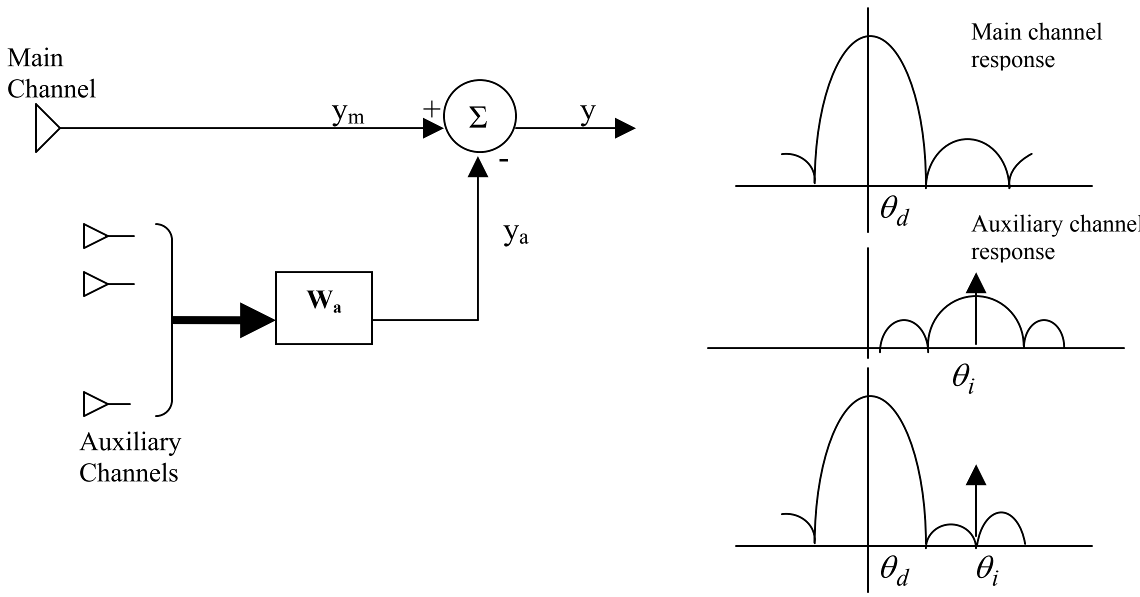

Figure 5.2 Sidelobe canceller beamforming

If the responses to the interferers of both the channels are similar then the overall response of the system will be zero, which can result in white noise. Therefore the weights are chosen to trade off interference suppression for white noise gain by minimizing the expected value of the total output power. Therefore the criteria can be expressed mathematically as follows;

$$\underbrace{\min}_{w_a} E\{\left| y_m - w_a^H x_a \right|^2 \tag{5.17}$$

The optimum weights w_a which correspond to the sidelobe canceller's adaptive component were found to be

$$w_a = R_a^{-1} r_{ma} \tag{5.18}$$

$R_a = E\{x_a x_a^H\}$ is the auxiliary array correlation matrix and the vector r_{ma} is the cross correlation between auxiliary array elements and the main array. This technique is simple in operation but it is mainly effective when the desired signal is weaker compared to the interfering signals since the stronger the desired signal gets (relatively), its contribution to the total output power increases and in turn increases the cancellation percentage. It can even cause the cancellation of the desired signal.

5.2.2 Linearly Constrained Minimum Variance (LCMV)

Most of the beamforming techniques discussed require some knowledge of the desired signal strength and also the reference signal. These limitations can be overcome through the application of linear constraints to the weight vector. LCMV spatial filters are beamformers that choose their weights so as to minimize the filter's output variance or power subject to constraints. This criterion together with other constraints ensure signal preservation at the location of interest while minimizing the variance effects of signals originating from other locations.

In LCMV beamforming the expected value of the array output power is minimized, i.e.

$E\{|y|^2\} = w^H R_x w$ is minimized subject to $C^H w = f$; (5.19)

where R_x denotes the covariance matrix of $x(t)$, C is the constraint matrix which contains K column vectors and f is the response vector which contains K scalar constraint values. The solution to the above equation using Lagrange multipliers gives the optimum weights as

$$w_{opt} = R_x^{-1} C (C^H R_x^{-1} C)^{-1} f \tag{5.20}$$

This beam forming method is flexible and does not require reference signals to compute optimum weights but it requires computation of a constrained weight vector C.

5.2.3 Null steering beamforming

Unlike other algorithms null steering algorithms do not look for the signal presence and then enhance it, instead they examine where nulls are located or the desired signal is not present and minimize the output signal power. One technique based on this approach is to minimize the mean squared value of the array output while constraining the norm of the weight vector to be unity.

$$\min_{w} w^H R w \quad \text{subject to } w^H A w = 1 \tag{5.21}$$

The matrix A, a positive-definite symmetric matrix, serves to balance the relative importance of portions of the weight vectors over others. The optimum weight vector must satisfy the following equation;

$$Rw = -\lambda A w \tag{5.22}$$

5.2.4 Sample Matrix Inversion (SMI) Algorithm:

In this algorithm the weights are chosen such that the mean-square error between the beamformer output and the reference signal is minimized. The mean square error is given by

$$E\left[\{r(t) - w^H x(t)\}^2\right] = E[r^2(t)] - 2w^H R_r + w^H R_m w \tag{5.23}$$

$x(t)$ is the array output at time t; $r(t)$ is the reference signal; $R_m = E[x(t)x^H(t)]$ is the signal covariance matrix. $R_r = E[r(t)x(t)]$, defines the covariance between the reference signal and the data signal. The weight vector, for which the above equation becomes minimum, is obtained by setting its gradient vector with respect to w, to zero, i.e.

$$\nabla_w \{E\left[\{r(t) - w^H x(t)\}^2\right]\} = -2R_r + 2R_m w = 0 \tag{5.24}$$

Therefore,

$$w_{opt} = R_m^{-1} R_r \tag{5.25}$$

The optimum weights can be easily obtained by direct inversion of the covariance matrix. This algorithm requires a reference signal and is computational intensive. It is definitely faster than LMS.

5.2.5 Constant Modulus Algorithm (Blind adaptive beamforming)

The configuration of CMA adaptive beamforming is the same as that of the SMI system discussed above except that it requires no reference signal. It is a gradient-based algorithm that works on the theory that the existence of interference causes changes in the amplitude of the transmitted signal, which otherwise has a constant envelope (modulus). The weight updates are obtained by minimizing the positive mean cost function;

$$J_n = \frac{1}{2} E[(|y(n)|^2 - y_0^2)^2] \tag{5.26}$$

The weight updates are given by,

$$w(n+1) = w(n) - \mu g(w(n)) \tag{5.27}$$

$y(n)$ is the array output after the nth iteration, y_0 is the amplitude of the modulus of the desired signal in the absence of interference and $g(w(n))$ denotes an estimate of the cost function. The CMA algorithm may not always converge. This problem is overcome by having additional information about the desired signal.

5.2.6 Least Mean Squares (LMS):

This algorithm like the preceding one requires a reference signal and it computes the weight vector using the equation

$$w(n+1) = w(n) + \mu x(n)[d^*(n) - x^H(n)w(n)] \tag{5.28}$$

w(n+1) denotes weight computed at (n+1)th iteration.

Where μ is the gain constant that controls the rate of adaptation, i.e. how fast and how close the estimated weights approach the optimal weights. The convergence of the algorithm depends upon eigenvalues of R (the array correlation matrix), the array correlation matrix. In a digital system, the reference signal is obtained by periodically transmitting a training sequence that is known to a receiver, or using the spread code in the case of a direct-sequence CDMA system. The LMS algorithm described here is a basic structure for most dynamic adaptive algorithms. This method requires information about a reference signal.

5.2.7 Recursive Least Square algorithm (RLS)

As discussed above, the convergence of the LMS algorithm depends upon the eigenvalues of R. If R leads to a large spread, the algorithm converges slowly. This problem is solved here by replacing the step gradient size μ with a gain matrix $R^{-1}(n)$ at the nth iteration, producing the following weight equation,

$$w(n) = w(n-1) - R^{-1}(n)x(n)\varepsilon^{*}(w(n-1)) \tag{5.29}$$

where $R(n)$ is given by

$$R(n) = \delta_0 R(n-1) + x(n)x^{H}(n) \tag{5.30}$$

$$= \sum_{k=0}^{n} \delta_0^{n-k} x(k)x^{H}(k) \tag{5.31}$$

where, δ_0, a real scalar smaller than but close to 1, is used for exponential weighting of past data. ε is the error signal. The RLS algorithm does not require any matrix inversion computations as the inverse correlation matrix is computed directly. It requires reference signal and correlation matrix information. It is almost ten times faster compared to LMS.

5.3 Direction of Arrival (DOA) Algorithms

For the beamformer to steer the radiation in a particular direction and to place the nulls in the interfering directions the direction of arrival has to be known beforehand. The Direction of arrival algorithms does exactly the same; they work on the signal received at the output of the array and computes the direction of arrivals of all the incoming signals. Once the angle information is known it is fed into the beamforming network to compute the complex weight vectors required for beam steering. Some of the DOA algorithms are discussed in the following sections.

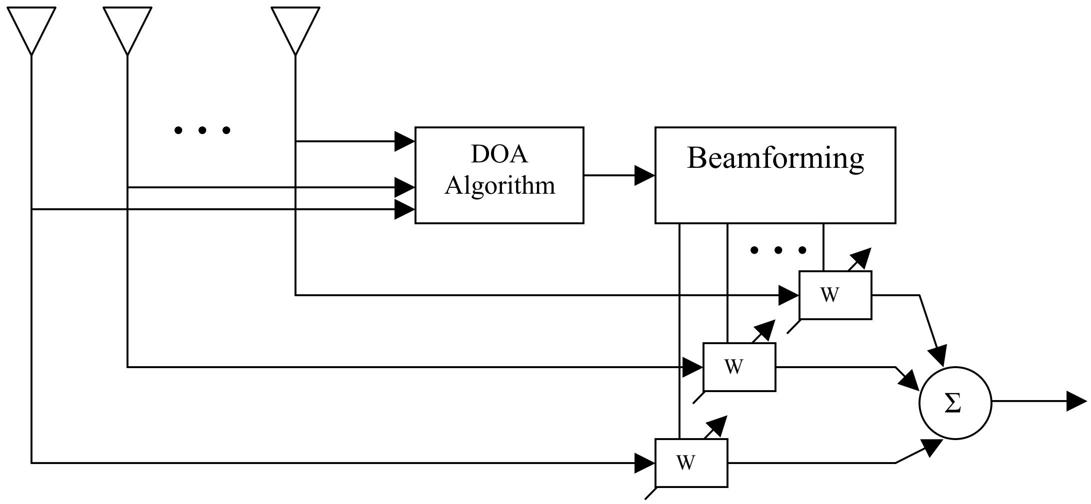

Figure 5.3 Beamforming setup with Direction of arrival estimation

5.3.1 MUSIC (MUltiple SIgnal Classification)

Consider a N-element linear array that detects M signals impinging on it whose directions of arrival need to be known. From the previous discussion we know that the received signals at the output of the array have the following form.

$$x(t) = A(\theta)S(t) + n(t) \tag{5.32}$$

or in matrix notation it can be represented as,

$$X = AS + n \tag{5.33}$$

where S is the signal vector, A is the array propagation vector and n is the noise vector with zero mean and σ^2 variance.

The $N \times N$ covariance matrix is given by,

$$R_x = E\{XX^*\} = AE\{SS^*\}A^* + E\{nn^*\} \tag{5.34}$$

$$= AP^* A + \sigma^2 I \tag{5.35}$$

where $P = E\{SS^*\}$ (5.36)

$$AP^* A = R_x - \sigma^2 I \tag{5.37}$$

when the number of signals M is less than N then $AP^* A$ is singular and has a rank less than N. The eigenvalues of R_x can be found by,

$$\left|AP^* A\right| = \left|R_x - \sigma^2 I\right| = 0 \tag{5.38}$$

The eigenvectors of R_x must satisfy,

$$R_x e_i = \sigma^2 I e_i \tag{5.39}$$

$$(R_x - \sigma^2 I) e_i = 0 \tag{5.40}$$

where e_i is the eigenvector and i varies from 1 to N-M.

Let the noise eigenvector be defined as E_N such that,

$$(s - \sigma^2 I) E_N = 0 \tag{5.41}$$

or,

$$APA^* E_N = 0 \tag{5.42}$$

Based upon this approach, the psuedospectrum $P(\theta)$ is given by,

$$P(\theta) = \frac{1}{A(\theta)^* E_N E_N{}^* A(\theta)} \tag{5.43}$$

when the psuedospectrum $P(\theta)$ is plotted, peaks appear at the angles of arrival of the incident signals.

5.3.2 ESPRIT (Estimation of Signal Parameters via Rotational Invariance Technique)

ESPIRIT is one of the most efficient and robust methods for DOA estimation. It uses two arrays in the sense that the second element of each pair is displaced by the same distance in the same direction relative to the first element. It is not required to have two separate arrays but can be realized using a single array by being able to select a subset of elements.

Let the array signals received by the two arrays be denoted by $x(t)$ and $y(t)$ such that

$$x(t) = As(t) + n_x(t) \tag{5.44}$$

$$y(t) = A\phi s(t) + n_y(t) \tag{5.45}$$

A is a $K \times M$ matrix; where M is the number of steering vectors produced by N elements of the array. $n_x(t)$ and $n_y(t)$ denotes the noise induced at the elements of the two arrays. Now, by using the available methods, the numbers of directional sources, M , are estimated based on principles such as Akaike's information criterion (AIC) and Minimum description length (MDL). Two matrices U_x and U_y are formed which denote the M eigenvectors corresponding to the largest eigenvalues of the two array correlation matrices R_{xx} & R_{yy} (Array correlation matrices). The eigenvectors of the following 2M by 2M matrix are obtained and are denoted by

$$V = \begin{bmatrix} U_x^H \\ U_y^H \end{bmatrix} \begin{bmatrix} U_x U_y \end{bmatrix} \tag{5.46}$$

Once the eigenvector V is obtained its eigenvalues $\lambda_m, m = 1,.....,M$ are computed.

Now the DOA is given by

$$\theta_m = \cos^{-1}\{\frac{Arg(\lambda_m)}{2\pi\Delta_0}\}, m = 1,.......,M \tag{5.47}$$

Δ_0 is the element separation in terms of wavelengths. Other variations of ESPRIT include beam-space ESPRIT, resolution-enhanced ESPRIT multiple invariance ESPRIT and higher order ESPRIT.

The **ESPIRIT DOA** estimation technique is found to be more robust and faster when compared to **MUSIC** technique. The computation is also less complex comparatively. However, **ESPIRIT** cannot handle correlated sources. In the next chapters the LMS and SMI algorithms will be studied in detail with simulations.

In the next chapters the LMS and SMI algorithms will be studied in detail with simulations.