INTERFERENCE CANCELLATION USING A HOPFIELD NEURAL NETWORK

A. M. Haimovich
B. Even-Or
AEL Industries
305 Richardson Rd.
Lansdale, PA 19446
(215) 822-2929

Abstract

This paper investigates the use of a Hopfield neural network for interference cancellation. It is suggested to use the neural network to estimate the signal subspace that contains the interferences. The array weight vector is constructed orthogonal to this subspace. It is shown that this array provides effective interference cancellation for narrowband sources.

1. Introduction

A large number of algorithms have been developed for beamforming with interference cancellation. The problem has been traditionally solved using matrix manipulation techniques.
Methods based on inversion of the spatial covariance matrix, such as the maximum likelihood (ML), require $o(N^3)$ operations. The ML method provides optimal signal-to-noise ratio and accurate pattern zeros for multiple interference sources spaced wider than a beamwidth. However, the performance of those methods is dependent on the interference-to- background-noise ratio and the spatial distribution of the sources. For interference sources spaced closer than a beamwidth the pattern zeros will be generally displaced from the true source bearings. Furthermore, the array will not necessarily allocate a zero to each source. For these cases, methods based on the eigen analysis of the covariance matrix improve the interference cancellation. The eigen analysis requires $o(N^4)$ operations, which makes real-time operation prohibitively expensive for most applications. With advances in VLSI technology, parallel architectures have been proposed to make real-time applications possible.

Recently much interest has been focused on using neuron-like networks to solve complex optimization problems. Those networks are capable of identifying which of the exemplars used to "train" the network is best represented by a noisy input. Neural networks have been applied to direction finding problems. In particular, a number of researchers have used the Hopfield net to obtain super-resolution angle of arrival estimators. Since the neural network solves a least-squares problem, it can resolve uncorrelated as well as correlated sources. This feature makes the neural network particularly attractive in situations where multipath reflections are a problem.

In this paper we suggest the usage of a Hopfield network to solve the interference cancellation problem. The approach taken may be viewed as an extension of the concept of Eigen-canceler developed by the author, [1]. The neural processor substitutes for one of the building blocks of the Eigen-canceler, namely the algorithm that performs the eigen-analysis of the array covariance matrix. A functional block diagram of the interference canceler is shown in Figure 1. Prior to processing by the neural network, it is necessary to remove the desired signal from the data to preclude its cancellation with the interference. This is achieved by a blocking transformation. The resulting modified data vector is input to the Hopfield neural net. The neural net output is used to estimate the interference subspace. To provide the beamforming and interference cancellation functions in a coordinated manner, the array weight vector is then constructed to meet the following requirements:

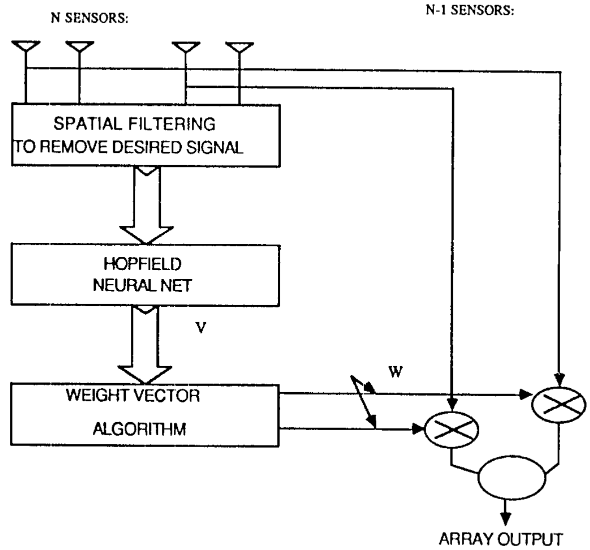

Figure 1. Functional Diagram of a Neural Network Interference Canceler

1. Cancel the interference, namely, the weight vector is orthogonal to the interference subspace
2. Provide fixed gain to the desired signal
3. Optimize the signal to background noise ratio

2. Signals, Interferences and Noise

Our model comprises a collinear array with N equally spaced elements and two independent sources: a desired source with known angle of arrival and an interference to be canceled. The assumption of two sources is made for simplicity. To handle multiple signals, the network has to be expanded to account for the interactions between the position vectors spanning the signal subspace, [2]. Plane waves originating from the sources impinge on the array at distinct angles $\{\theta_l, l = 1,2\}$. The signals are assumed narrowband, zero-mean stochastic, processes of the form $a_l e^{j\omega_0 t+\phi_l(t)}$, $l = 1,2$, where a_l, ω_0 and $\phi(t)$ are the i-th signal source amplitude, frequency and phase, respectively. For convenience, signals will be denoted by their complex envelopes, s_l, $l = 1,2$.

CH2868-8/90/0000-0503$1.00 © 1990 IEEE

The array output is sampled, and the k-th snapshot is given by:

$$\bar{\boldsymbol{x}}(k) = s_0(k)\bar{\boldsymbol{d}}_0 + s_1(k)\bar{\boldsymbol{d}}_1 + \bar{\boldsymbol{\nu}}(k) \qquad (1)$$

where the subscripts "0" and "1" refer to the desired signal and the interference respectively. $\bar{\boldsymbol{\nu}}$ is the measurement noise which is assumed white Gaussian and uncorrelated from sensor to sensor. The position vectors $\bar{\boldsymbol{d}}_{0,1}$ are given by:

$$\bar{\boldsymbol{d}}_{0,1} = [1 \; e^{ju_{0,1}} \; \cdots \; e^{j(N-1)u_{0,1}}]^T \qquad (2)$$

$u_{0,1}$ is related to the source angle of arrival by $u_{0,1} = 2\pi l \sin\theta_{0,1}/\lambda$; l is the interelement spacing and λ the wavelength emanating from the sources.

The signal blocking is intended to remove the desired signal from the data to preclude its cancellation. The theory for the signal blocking transformation is given in full detail in [1]. This transformation has in its nullspace the position vector of the desired signal, and while modifying the interference position vector, it preserves its planar feature. It is important to point out that while the weight vector is derived from the modified data, the preservation of the plane waveform enables the application of this weight vector directly to the sensors. The signal blocking transformation $\boldsymbol{C}$ has the following properties:

1. $\boldsymbol{C}$ is $(N, N-1)$ and has full column rank.
2. The desired signal position vector is in the nullspace of C.
$$\boldsymbol{C}\bar{\boldsymbol{d}}_0 = 0 \qquad (3)$$
3. Other position vectors are preserved planar though they may be rotated by C.
4. $$\boldsymbol{C}\bar{\boldsymbol{d}}_1 = \alpha \boldsymbol{d}_1 \qquad (4)$$

where $\boldsymbol{d}_1$ is defined similarly to $\bar{\boldsymbol{d}}_1$ except it is $(N-1)$-dimensional rather than N-dimensional. Namely, $\bar{\boldsymbol{d}}_1 = [1 \; e^{ju_1} \; \cdots \; e^{j(N-2)u_1}]^T$. Using those properties the modified data vector $\boldsymbol{x}$ can be written:

$$\begin{aligned} \boldsymbol{x}(k) &= \boldsymbol{C}\bar{\boldsymbol{x}}(k) \\ &= \alpha s_1 \boldsymbol{d}_1 + \boldsymbol{C}\bar{\boldsymbol{\nu}}(k) \end{aligned} \qquad (5)$$

Note that $\boldsymbol{x}(k)$ has a term related to the interference source and a term related to the noise.

3. Beamforming utilizing a Hopfield Network

Hopfield [3] suggested to solve optimization problems using a network of highly interconnected non-linearities. In particular, he showed how to relate the network to an energy function. Rastogi et. al., [2] and others applied Hopfield's methods to estimate the angle of arrival of a planar waveform. Our objective is to apply the same methods in the context of the interference cancellation problem. The modified data vector $\boldsymbol{x}$ consists of a planar waveform (the modified interference position vector) and noise. The following cost function is minimized utilizing a p- neuron Hopfield network:

$$E = \frac{1}{K}\sum_{k=1}^{K} \mid \boldsymbol{x}(k) - [\boldsymbol{P}_1\boldsymbol{x}(k) \; \ldots \; \boldsymbol{P}_p\boldsymbol{x}(k)]\boldsymbol{v} \mid^2 \qquad (6)$$

where K is the number of data snapshots, $\boldsymbol{P}_i = \boldsymbol{d}_i(\boldsymbol{d}_i^\dagger\boldsymbol{d}_i)^{-1}\boldsymbol{d}_i^\dagger$, $i = 1, \ldots, p$ is a projection matrix that projects the modified data vector $\boldsymbol{x}(k)$ onto the signal direction vector $\boldsymbol{d}_i = \boldsymbol{d}(\theta_i)$. A stereotype position vector $\boldsymbol{d}(\theta_i)$ is associated to each of the p neurons of the network. The state vector v takes values between 0 and 1. The output states corresponding to the position vectors contained in the received data will take on values close to 1, while the other states will take on values close to 0. The number of neurons in the network determines the resolution in estimating the angle of arrival of the signal to be canceled.

Using the approach taken by Rastogi et. al., [2], the network weights are data dependent and computed from the relations:

$$\begin{aligned} T_{i,j} &= -2 \,\text{Real}\, \frac{1}{K}\sum_{k=1}^{K} \boldsymbol{x}^\dagger \boldsymbol{P}_i \boldsymbol{P}_j^\dagger \boldsymbol{x} \quad i,j = 1 \ldots p &(7) \\ T_{i,i} &= 0 &(8) \end{aligned}$$

The network bias inputs are given by:

$$I_i = \frac{1}{2} \,\text{Real}\, \frac{1}{K}\sum_{k=1}^{K} \boldsymbol{x}^\dagger \boldsymbol{P}_i \boldsymbol{x} \quad i = 1 \ldots p \qquad (9)$$

$$(10)$$

The neuron input/output relation is given by:

$$\boldsymbol{v}(k) = \frac{1}{2}[1 + \tanh(\frac{\boldsymbol{u}(k)}{\lambda})] \qquad (11)$$

where $\boldsymbol{u}$ and $\boldsymbol{v}$ are the p-dimensional input/output neuron vectors, and λ is a gain factor. The neurons' output is updated using the relation:

$$\boldsymbol{u}(k+1) = \boldsymbol{u}(k) + \boldsymbol{T}\boldsymbol{v} + \boldsymbol{I} \qquad (12)$$

For the case of a single interference source, and after convergence, one of the elements of $\boldsymbol{v}$ will take value 1, $v_S = 1$, while the rest will have values close to zero. This output state corresponds to the projection matrix $\boldsymbol{P}_S = \boldsymbol{d}_S(\boldsymbol{d}_S^\dagger\boldsymbol{d}_S)^{-1}\boldsymbol{d}_S^\dagger$. If the network converged to the global minimum then $\boldsymbol{d}_S$ is the stereotype position vector closest to the interference position vector $\boldsymbol{d}_1$. Ignoring noise effects, we have:

1. If $\boldsymbol{d}_S = \boldsymbol{d}_1$, $v_S = 1$, all other output states equal to zero:
$$\begin{aligned} E &= \mid \gamma \mid^2 \mid \boldsymbol{d}_1 - \boldsymbol{P}_S\boldsymbol{d}_1 \mid^2 \\ &= 0 \end{aligned} \qquad (13)$$
where γ is a complex scalar. The cost function is minimized and the output state $v_S = 1$ points correctly to the interference position vector.
2. If $\boldsymbol{d}_S \neq \boldsymbol{d}_1$, $v_S = 1$, all other output states equal to zero:
$$\begin{aligned} E &= \mid \gamma \mid^2 \mid \boldsymbol{d}_1 - \boldsymbol{P}_S\boldsymbol{d}_1 \mid^2 \\ &= \mid \gamma \mid^2 \mid \boldsymbol{d}_1 - \rho\boldsymbol{d}_S \mid^2 \\ &= \mid \gamma \mid^2 (N-1)(1 - \mid \rho \mid^2) \end{aligned} \qquad (14)$$
where $\rho = (d_S^\dagger d_S)^{-1} d_S^\dagger \boldsymbol{d}_1$. Note that $d_S^\dagger d_S = (N-1)$. The cost function is minimized by the largest $\mid \rho \mid$. The largest $\mid \rho \mid$ is achieved by the position vector $\boldsymbol{d}_S$ closest to the interference position vector $\boldsymbol{d}_1$.

There are not known conditions that ensure convergence to the global minimum; however, we found that, in practice, the state vector converged to the global minimum in the majority of cases.

Define the vector:

$$\boldsymbol{Q}(\boldsymbol{x}) = \frac{1}{K}\sum_{k=1}^{K}[\boldsymbol{P}_1\boldsymbol{x}(k)\ \dots\ \boldsymbol{P}_p\boldsymbol{x}(k)]\boldsymbol{v} \tag{15}$$

Then, after convergence, we have :

$$\boldsymbol{Q}(\boldsymbol{x}) \;=\; \frac{1}{K}\sum_{k=1}^{K}\boldsymbol{P}_S\boldsymbol{x} \tag{16}$$

After some calculations it can be shown that:

$$\boldsymbol{Q}(\boldsymbol{x}) = \beta\boldsymbol{d}_1 + \boldsymbol{f} \tag{17}$$

where β is a complex scalar and $\boldsymbol{f}$ is a noise vector. We use the vector $\boldsymbol{Q}(\boldsymbol{x})$, which can be calculated from the data and the network state vector $\boldsymbol{v}$, as the estimate to the interference position vector, and construct an array weight vector $\boldsymbol{w}$ orthogonal to $\boldsymbol{Q}$. If $\boldsymbol{Q}$ is a good estimate of $\boldsymbol{d}_1$, then the weight vector will place a null in the array pattern in the direction of the interference.

To simultaneously minimize the output noise power, cancel the interference, and set the gain for the desired signal, the following optimization is used:

$$\min_{\boldsymbol{w}}\ \boldsymbol{w}^{\dagger}\boldsymbol{w} \quad \text{subject to} \quad \boldsymbol{w}^{\dagger}\boldsymbol{Q}(\boldsymbol{x}) = 0 \quad \text{and} \quad \boldsymbol{w}^{\dagger}\boldsymbol{d}_0 = g \tag{18}$$

where g is the required gain to the desired signal. Using Lagrange multipliers to calculate the optimal weight vector we get:

$$\boldsymbol{w} = \boldsymbol{d}_0 - \frac{\boldsymbol{Q}^{\dagger}\boldsymbol{d}_0}{\boldsymbol{Q}^{\dagger}\boldsymbol{Q}}\boldsymbol{Q} \tag{19}$$

Since $\boldsymbol{Q}$ is a function of the data $\boldsymbol{x}$, so is the weight vector $\boldsymbol{w}$. This weight vector is now copied back to the sensors. Note that, while the array has N elements, $\boldsymbol{w}$ is only $(N-1)$-dimensional. The degree of freedom missing from the weight vector has been used by the signal blocking transformation.

4. Results and Discussion

To evaluate the proposed interference cancellation method, we considered a four element array and carried out a series of computer simulations. The model consisted of an interference source at $\sin\theta = 0.4$ degrees off broadside, and a desired source at broadside. The two sources emitted narrowband waveforms with independent (between sources and between samples) phase distributed uniformly in the range $(-\pi, \pi)$. The desired signal was 10 dB above the noise at each element, while the interference-to-noise ratio was varied to examine its effect on the canceler's performance. The noise was modeled as complex Gaussian with zero-mean and unity variance. $K = 20$ snapshots were averaged for the computation of the network weights and the vector $\boldsymbol{Q}$.

Figure 2(a) through (d) shows the adapted array pattern for increasing interference-to-noise ratios. The ordinate is the sine of the angle of look, and the abscissa is the array gain in linear arbitrary units. The neural processor places accurate nulls in the

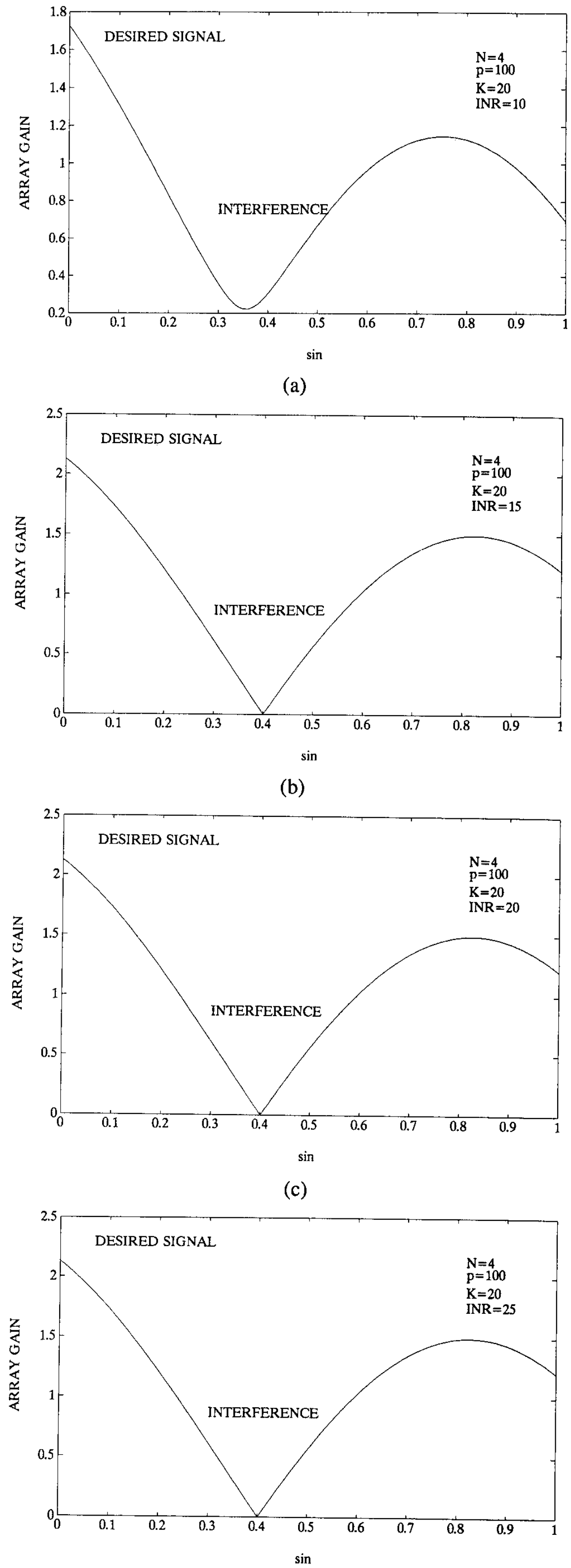

Figure 2. Adapted Array Patterns and Interference Cancellation Time Responses for Different Interference-to-Noise Ratios

direction of the interference, except for the low INR = 10 dB case, where the null is offset. By comparing Figure 2(a) with Figure 3, the quiescent (before adaptation) array pattern, it is observed that even for the low interference power case there is some interference cancellation. Figure 2(e) through (h) shows the array gain to the desired signal and the interference during the convergence process. The linear constraint in (18) preserves the desired signal during the process. The higher the interference-to-power, the closer is $\boldsymbol{Q}$ to the actual interference position vector, hence the better interference cancellation. While for INR = 10 dB, the interference is suppressed only by 5 dB, for INR = 25 dB, the suppression is better than 40 dB.

5. Conclusions

In this paper the application of a Hopfield neural network to the interference cancellation problem has been proposed. The case of a single desired signal and single interference has been treated. Results presented here show that the array places an accurate null in the direction of the interference.

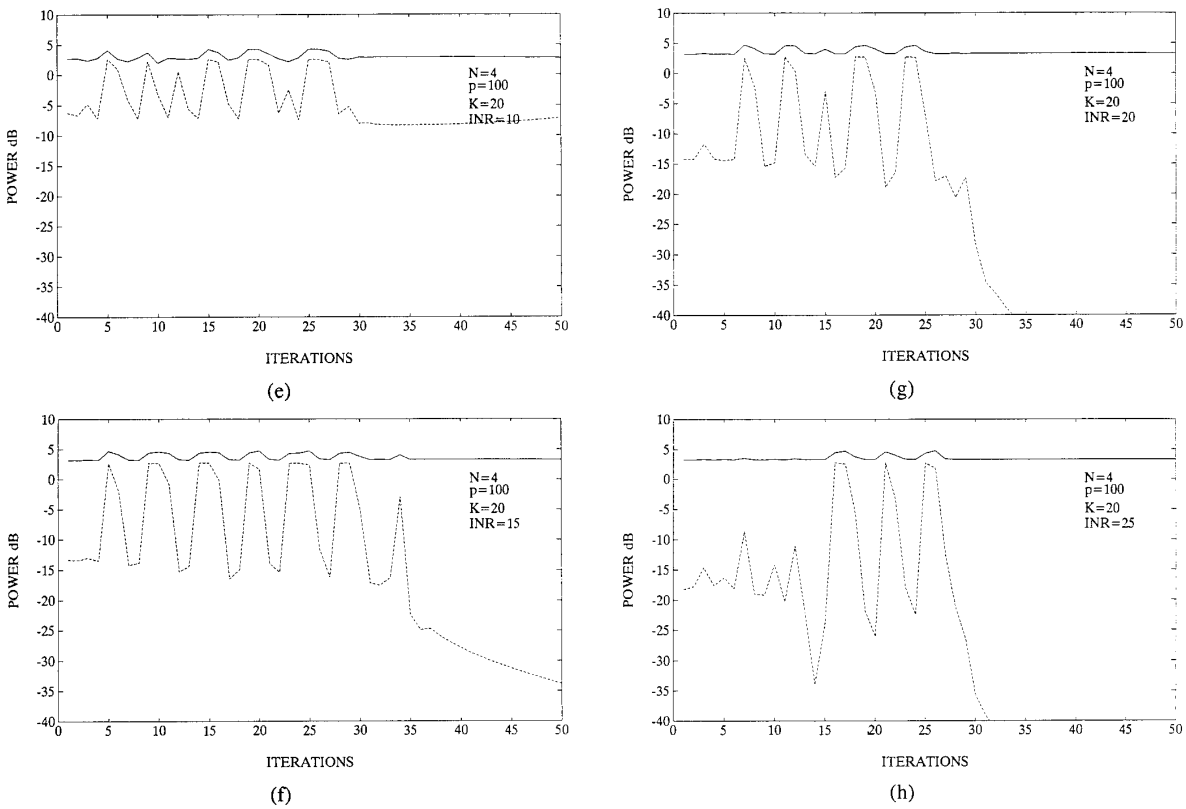

Figure 2. Adapted Array Patterns and Interference Cancellation
Time Responses for Different Interference-to-Noise Ratios - Continued

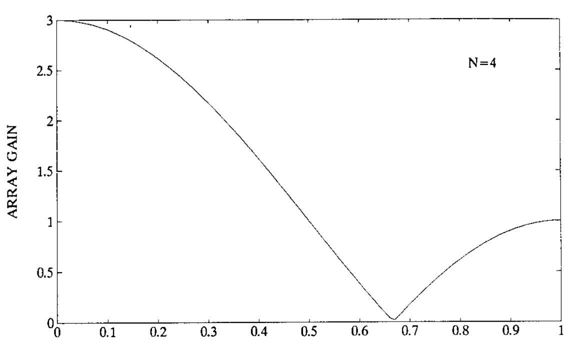

Figure 3. Quiescent Array Pattern

References

[1] A. Haimovich, "Eigenanalysis Techniques for Cancelling Narrowband Interferences," PhD dissertation, University of Pennsylvania, 1989.

[2] R. Rastogi et. al., "Array Signal Processing with Interconnected Neuron-like Elements," Proceedings IEEE ICASSP'87, pp.2328- 2331, 1987.

[3] J. J. Hopfield, "Neural Networks and Physical Systems with Emergent Collective Computational Geometry," Proceedings Natural Academy of Sciences, USA, Vol. 79, pp.2554-2558, Apr. 1982.