A Recursive Programmable Canonical Correlation Analyzer*

Mark F. Kahn
University of California, Davis
E-Systems, Inc., Greenville Division
Greenville, TX 75403

Matthew A. Mow
University of California, Davis
E-Systems, Inc., Greenville Division
Greenville, TX 75403

William A. Gardner
University of California, Davis
Davis, CA 95616

Thomas E. Biedka
E-Systems, Inc., Greenville Division
Greenville, TX 75403

August 31, 1994

Abstract

The Programmable Canonical Correlation Analyzer (PCCA) is modified to include recursion, feedback and training set constraints for improved blind adaptive spatial filtering. The development presented here includes the establishment of appropriate optimization criteria, optimal weight solutions (when tractable), and efficient power method implementations for the new PCCA techniques. The resulting family of adaptive spatial filters is shown to include previously established beamforming techniques, as well as several new processors. The performance of these techniques is evaluated by monte carlo simulation and characterized in terms of output SINR and convergence behavior. In many cases the new PCCA techniques outperform established techniques at the expense of increased computational burden.

I Introduction

The Programmable Canonical Correlation Analyzer (PCCA) has recently been proposed as a general signal processing structure for blind adaptive spatial filtering [4, 10, 11]. The PCCA computes the canonical correlation [3, 7] between a data-derived training signal set and a spatial filter output in order to permit extraction of signals exhibiting preselected statistical properties. This is accomplished through the joint adaptation of a pair of linear combiners operating on the input data and training signal sets respectively. In this paper, the PCCA structure is modified to permit greater flexibility in the selection and control of the training signal set. In particular, two modifications are developed. The first permits the user to incorporate the spatial filter output within the training set in a recursive or feedback mode. The second modification permits the user to place linear constraints on the training set weights in order to ensure exploitation of desired signal properties. With these modifications a number of established blind adaptive spatial filtering algorithms (*e.g.*, LS-CMA [1] and Phase-SCORE [9]) are now able to be presented within a general PCCA framework.

Prior to development of the modified PCCA a brief review of the basic PCCA and related adaptive spatial filtering techniques is presented. These techniques are characterized in terms of their design optimization criteria, adaptation procedures, optimal solutions (if tractable), and weight update implementations. To facilitate the comparison of these techniques a common array model and notation convention is introduced.

I.A Model and Notation Conventions

Given an array of M sensors the received data is represented by the narrowband model

$$\mathbf{x}(k) = \mathbf{a}s(k) + \mathbf{n}(k), \tag{1}$$

where $\mathbf{a}$ is the array steering vector, $s(k)$ is the signal of interest, and $\mathbf{n}(k)$ is the undesired interference and noise. The vectors $\mathbf{x}(k)$, $\mathbf{a}$, and $\mathbf{n}(k)$ are complex $M \times 1$ vectors. The output of a linear combiner applied to the received array data is given by

$$y(k) = \mathbf{w}^H\mathbf{x}(k), \tag{2}$$

where $\mathbf{w}$ is an $M \times 1$ vector of complex combiner weights, and $(\cdot)^H$ is the complex transposition operation. Similarly we denote a training signal by

$$r(k) = \mathbf{c}^H\mathbf{z}(k), \tag{3}$$

*This work supported by Rome Laboratories RL/IRAA.

where $\mathbf{z}(k)$ is an arbitrary function (or transformation) of the array data vector and/or beamformer output given by

$$\mathbf{z}(k) = f\left[\mathbf{x}(\cdot), y(\cdot)\right](k). \tag{4}$$

In general, $\mathbf{z}(k)$ can be any complex $P \times 1$ vector, and $f[\cdot]$ is an arbitrary linear or nonlinear mapping, with or without memory. Correlation matrices for pairs of complex vector-valued sequences are denoted by

$$\mathbf{R_{uv}} \triangleq \left\langle \mathbf{u}(k)\mathbf{v}^H(k)\right\rangle,$$

where $\langle\cdot\rangle$ is the time averaging operation.

I.B The PCCA

The PCCA processing structure is presented in Figure 1. The PCCA seeks to minimize the mean (time averaged) squared error (MSE) between linear estimates of the common terms shared by the two data sets $\mathbf{x}(k)$ and $\mathbf{z}(k)$. The optimization criterion can be expressed as,

$$\min_{\mathbf{W},\mathbf{C}} \left\langle \left|\mathbf{W}^H\mathbf{x}(k) - \mathbf{C}^H\mathbf{z}(k)\right|^2\right\rangle, \tag{5}$$

subject to the constraints,

$$\mathbf{W}^H\mathbf{R_{xx}}\mathbf{W} = \mathbf{I}, \tag{6}$$
$$\mathbf{C}^H\mathbf{R_{zz}}\mathbf{C} = \mathbf{I}, \tag{7}$$

where $\mathbf{I}$ is the identity matrix. The optimal $\mathbf{W}$ and $\mathbf{C}$ combiner weights are given by the dominant eigenvector solutions of the following eigensystem of equations:

$$\left[\mathbf{R_{xx}^{-1}R_{xz}R_{zz}^{-1}R_{xz}^{H}}\right]\mathbf{W} = \mathbf{W}\Lambda \tag{8}$$
$$\left[\mathbf{R_{zz}^{-1}R_{xz}^{H}R_{xx}^{-1}R_{xz}}\right]\mathbf{C} = \mathbf{C}\Lambda. \tag{9}$$

These weights can be directly determined using a closed form eigenequation solution or by the following iterative *alternating block power method* [1]

$$\mathbf{C}_k = \mathbf{R_{zz}^{-1}R_{xz}^{H}}\mathbf{W}_{k-1} \tag{10}$$
$$\mathbf{W}_k = \mathbf{R_{xx}^{-1}R_{xz}}\mathbf{C}_k, \tag{11}$$

with an appropriate Gram-Schmidt Orthogonalization [5, 12] applied at each iteration k. Observe that $\mathbf{C}_k^H\mathbf{x}(n)$ is the minimum MSE estimate of $\mathbf{W}_{k-1}^H\mathbf{z}(n)$ and $\mathbf{W}_k^H\mathbf{x}(n)$ is the minimum MSE estimate of $\mathbf{C}_k^H\mathbf{z}(n)$, Thus, we are using a bootstrapping method to jointly optimize the signal estimate $\mathbf{W}^H\mathbf{x}(n)$ and the training signal $\mathbf{C}^H\mathbf{z}(n)$.

[1] See Appendix A for a summary of power methods [5, 12].

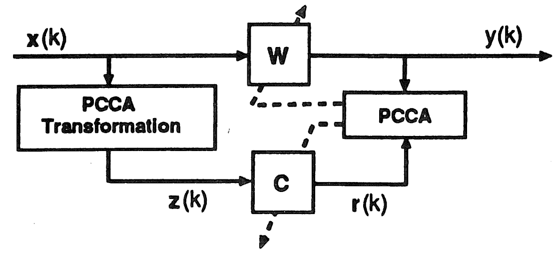

Figure 1: The PCCA processing structure.

I.C Property Exploiting Techniques

Many established adaptive spatial filtering techniques are closely related to the PCCA. Of particular interest in this paper are techniques developed for exploitation of either modulus or cyclostationarity properties. Three have been selected for performance comparison with the PCCA and include the LS-CMA, Cross-SCORE and Phase-SCORE algorithms. A brief summary of these techniques follows.

The LS-CMA [1] seeks to minimize the constant modulus (1-2) cost function

$$\min_{\mathbf{w}} \left\langle \left|\, |y(k)| - \delta\right|^2\right\rangle,$$

where δ is typically taken to be unity. A training signal is generated by the complex limiting operation

$$r(k) = \frac{y(k)}{|y(k)|},$$

and the combiner weight is determined by the block update

$$\mathbf{w} = \mathbf{R_{xx}^{-1}R_{xr}}.$$

The Cross-SCORE technique [2] is designed to maximize the cross-correlation coefficient of the observed data and training signal,

$$\max_{\mathbf{w},\mathbf{c}} \frac{|R_{yr}|^2}{[R_{yy}R_{rr}]} = \max_{\mathbf{w},\mathbf{c}} \frac{\left|\mathbf{w}^H\mathbf{R_{xr}}\right|^2}{[\mathbf{w}^H\mathbf{R_{xx}}\mathbf{w}]\,[\mathbf{c}^H\mathbf{R_{xx}}\mathbf{c}]}$$

where the training signal is specified by

$$r(k) \triangleq \mathbf{c}^H\mathbf{x}(k-n)e^{-j2\pi\alpha k}.$$

From the well known maximization of generalized Rayleigh quotients, the optimal $\mathbf{w}$ and $\mathbf{c}$ weights are shown to be the dominant eigenvectors of the following generalized eigensystem:

$$\left[\mathbf{R_{xx}^{\alpha}}(n)\mathbf{R_{xx}^{-1}}\mathbf{R_{xx}^{\alpha H}}(n)\right]\mathbf{w} = \lambda\mathbf{R_{xx}}\mathbf{w}$$

$$\left[\mathbf{R_{xx}^{\alpha H}}(n)\mathbf{R_{xx}^{-1}}\mathbf{R_{xx}^{\alpha}}(n)\right]\mathbf{c} = \lambda\mathbf{R_{xx}}\mathbf{c},$$

where $\mathbf{R}_{\mathbf{xx}}^{\alpha}(n) = \langle \mathbf{x}(k)\mathbf{x}^H(k-n)e^{+j2\pi\alpha k}\rangle$ is the cyclic autocorrelation matrix of the received data. The dominant eigenvectors can be determined from a closed form solution, or by the alternating power method

$$\begin{aligned} \mathbf{c}_k &= \mathbf{R}_{\mathbf{xx}}^{-1}\mathbf{R}_{\mathbf{xx}}^{\alpha H}(n)\mathbf{w}_{k-1} \\ &= \mathbf{R}_{\mathbf{xx}}^{-1}\mathbf{R}_{\mathbf{xz}}^{H}\mathbf{w}_{k-1} \\ \mathbf{w}_k &= \mathbf{R}_{\mathbf{xx}}^{-1}\mathbf{R}_{\mathbf{xx}}^{\alpha}(n)\mathbf{c}_k \\ &= \mathbf{R}_{\mathbf{xx}}^{-1}\mathbf{R}_{\mathbf{xz}}\mathbf{c}_k, \end{aligned} \tag{12}$$

where $\mathbf{z}(k) \triangleq f[\mathbf{x}(k)] = \mathbf{x}(k-n)e^{-j2\pi\alpha k}$.

The Phase-SCORE [9] algorithm is an ad hoc technique with no predetermined optimization criterion[2]. The desired weight vector $\mathbf{w}$ is the dominant eigenvector of the eigenequation

$$\left[\mathbf{R}_{\mathbf{xx}}^{-1}\mathbf{R}_{\mathbf{xx}}^{\alpha}(n)\right]\mathbf{w} = \lambda\mathbf{w}.$$

This weight vector can be iteratively computed using the basic power method as follows,

$$\mathbf{w}_{k+1} = \left[\mathbf{R}_{\mathbf{xx}}^{-1}\mathbf{R}_{\mathbf{xx}}^{\alpha}(n)\right]\mathbf{w}_k.$$

II The RPCCA Processor

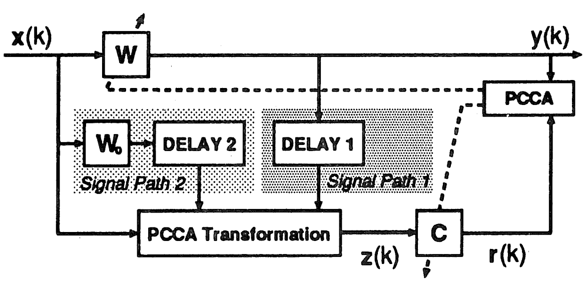

Figure 2: The modified PCCA structure.

The PCCA structure is now generalized to include recursion by incorporating two additional signal paths as indicated in Figure 2. The first PCCA modification to be considered utilizes signal path 1 only and thereby permits the beamformer output $\mathbf{y}(k)$ to be recursively incorporated within the training signal set. This structure is referred to as *recursive PCCA* (RPCCA). Note that the RPCCA operates on a 'block update' basis, *i.e.,* beamformer weight solutions are obtained for each new block of data. Therefore, the delay in signal path 1 is taken to be $B \triangleq$ *PCCA data block duration.* In order to distinguish between signals and weight vectors associated with different PCCA processing blocks, the subscript notation m and $m-1$ is introduced to indicate the current and previous processing blocks respectively.

The training signal set for the RPCCA is given by

$$\mathbf{r}(k) = \mathbf{C}_m^H\mathbf{z}(k), \tag{13}$$

where

$$\mathbf{z}(k) = \mathbf{f}\left[\mathbf{x}_m(\cdot), \mathbf{y}_{m-1}(\cdot)\right](k), \tag{14}$$

$$\mathbf{y}_{m-1}(\cdot) = \mathbf{W}_{m-1}^H\mathbf{x}_{m-1}(\cdot), \tag{15}$$

and $\mathbf{f}\left[\mathbf{x}_m(\cdot), \mathbf{y}_{m-1}(\cdot)\right](k)$ is referred to as the *RPCCA transformation.* The optimization criterion is expressed as

$$\min_{\mathbf{W}_m, \mathbf{C}_m} \left\langle \left|\mathbf{W}_m^H\mathbf{x}(k) - \mathbf{C}_m^H\mathbf{f}\left[\mathbf{x}_m(\cdot), \mathbf{y}_{m-1}(\cdot)\right](k)\right|^2 \right\rangle,$$

subject to the constraints $\mathbf{W}_m^H\mathbf{R}_{\mathbf{xx}}\mathbf{W}_m = \mathbf{I}$ and $\mathbf{C}_m^H\mathbf{R}_{\mathbf{zz}}\mathbf{C}_m = \mathbf{I}$. The solution to this optimization is given by the pair of dominant eigenvectors from the following system of eigenequations:

$$\left[\mathbf{R}_{\mathbf{xx}}^{-1}\mathbf{R}_{\mathbf{xz}}\mathbf{R}_{\mathbf{zz}}^{-1}\mathbf{R}_{\mathbf{xz}}^H\right]\mathbf{W}_m = \mathbf{W}_m\Lambda, \tag{16}$$

$$\left[\mathbf{R}_{\mathbf{zz}}^{-1}\mathbf{R}_{\mathbf{xz}}^H\mathbf{R}_{\mathbf{xx}}^{-1}\mathbf{R}_{\mathbf{xz}}\right]\mathbf{C}_m = \mathbf{C}_m\Lambda. \tag{17}$$

The dominant eigenvectors can be computed by directly solving the eigenequations, or by applying an appropriate power method. For the scalar case $z(k) \in \mathcal{C}^{1\times 1}$, the alternating power method yields the following iterative solution,

$$c_{m,k} = R_{zz}^{-1}\mathbf{R}_{\mathbf{xz}}^H\mathbf{w}_{m,k-1}, \tag{18}$$

$$\mathbf{w}_{m,k} = \mathbf{R}_{\mathbf{xx}}^{-1}\mathbf{R}_{\mathbf{xz}}c_{m,k}, \tag{19}$$

where the double subscript notation $a_{i,j}$ is used to denote the jth iteration within the ith processing block. If L multiple solutions are desired and the RPCCA transformation consists of P signals, the weights $\mathbf{w}_m$ and c_m can be extended to matrix form as $\mathbf{W}_m \in \mathcal{C}^{M\times L}$ and $\mathbf{C}_m \in \mathcal{C}^{P\times L}$. The RPCCA transformation vector is then $\mathbf{z}(k) \in \mathcal{C}^{P\times 1}$ and the resulting training set $\mathbf{r}(k)$ consists of L signals. The L most dominant eigenvector pairs of equations (16) and (17) provide the L minimal solutions to the optimization criterion. These eigenvector pairs can be computed directly through solution of the eigenequations or iteratively using the alternating block power method as follows,

$$\mathbf{C}_{m,k} = \mathbf{R}_{\mathbf{zz}}^{-1}\mathbf{R}_{\mathbf{xz}}^H\mathbf{W}_{m,k-1}, \tag{20}$$

$$\mathbf{W}_{m,k} = \mathbf{R}_{\mathbf{xx}}^{-1}\mathbf{R}_{\mathbf{xz}}\mathbf{C}_{m,k} \tag{21}$$

with appropriate Gram-Schmidt Orthogonalization.

[2]In Section II of this paper Phase-SCORE is shown to be a special case of the Recursive PCCA processor.

II.A Weight-Recursive PCCA

Two modifications to the RPCCA can be used to improve performance in many applications. The first seeks to maintain correlation between input and output sequences by applying the previous weight vector $\mathbf{w}_{m-1}$ to the *current* input data for use in generating the output training sequence[3]. This differs from RPCCA which processes the previous input data $\mathbf{x}_{m-1}$ to generate an output sequence $\mathbf{y}_{m-1}$. This modification is referred to as *weight-recursive PCCA* (WRPCCA) and is realized by the structure shown in Figure 2, utilizing signal path 2 only with $\mathbf{W}_o = \mathbf{W}_{m-1}$. The WRPCCA transformation is given by

$$\mathbf{z}(k) = \mathbf{f}\left[\mathbf{x}_m(\cdot), \hat{\mathbf{y}}_{m-1}(\cdot)\right](k), \quad (22)$$
$$\hat{\mathbf{y}}_{m-1}(k) = \mathbf{W}_{m-1}^H \mathbf{x}_m(k). \quad (23)$$

Using this $\mathbf{z}(k)$, equations (16) through (21) remain applicable for the WRPCCA. Note that this structure can exploit correlation between input and output signals which might have been uncorrelated in the RPCCA due to processing-block lags exceeding the signal correlation time.

The second RPCCA modification takes advantage of the iterative structure of the power method solution to permit an update of the training signal within a single processing block. At each power method iteration k the previously fixed block weight $\mathbf{w}_{m-1}$ is replaced by the iterative weight $\mathbf{w}_{m,k-1}$. This results in an output signal $\hat{\mathbf{y}}_m(k)$ that is recursively updated within a single PCCA processing block, presumably yielding a superior training signal estimate at each iteration. Alternatively, this approach can be interpreted as repetitively performing a single iteration WRPCCA power method while reusing the input data block. The notion of reapplying data in this manner is referred to as data recycling, hence the name WRPCCA/DR (weight-recursive PCCA with Data Recycling). In this case the block eigenequation solutions are not applicable, and the alternating power method solution must be modified as follows:

$$\begin{aligned} c_{m,k} &= R_{zz}^{-1}\mathbf{R}_{\mathbf{xz}}^H \mathbf{w}_{m,k-1}, \\ \mathbf{w}_{m,k} &= \mathbf{R}_{\mathbf{xx}}^{-1}\mathbf{R}_{\mathbf{xz}} c_{m,k}, \\ \hat{\mathbf{y}}_{m,k}(l) &= \mathbf{w}_{m,k}^H \mathbf{x}_m(l), \\ \mathbf{z}_k(l) &= \mathbf{f}\left[\mathbf{x}_m(\cdot), \hat{\mathbf{y}}_{m,k}(\cdot)\right](l). \end{aligned}$$

[3]This notion was similarly proposed in [1] for application in LS-CMA.

II.B Cyclic-RPCCA

The RPCCA can be designed to exploit cyclostationarity by utilizing the RPCCA transformation

$$\mathbf{f}\left[\mathbf{x}_m(\cdot), \mathbf{y}_{m-1}(\cdot)\right](k) = \mathbf{y}_{m-1}(k)e^{+j2\pi\alpha k}.$$

The relevent RPCCA signals are given by,

$$\begin{aligned} \mathbf{y}_{m-1}(k) &= \mathbf{W}_{m-1}^H \mathbf{x}(k)_{m-1} \\ \mathbf{z}(k) &= \mathbf{y}_{m-1}(k)e^{+j2\pi\alpha k} \\ \mathbf{r}(k) &= \mathbf{C}_m^H \mathbf{z}(k) \\ &= \mathbf{C}_m^H \mathbf{y}_{m-1}(k)e^{+j2\pi\alpha k}. \end{aligned}$$

The required PCCA correlation and cross-correlation matrices can be expressed as,

$$\begin{aligned} \mathbf{R}_{\mathbf{xz}} &= \mathbf{R}_{\mathbf{xx}}^{\alpha}\mathbf{W}_{m-1}, \\ \mathbf{R}_{\mathbf{zz}} &= \mathbf{W}_{m-1}^H \mathbf{R}_{\mathbf{xx}}\mathbf{W}_{m-1}, \\ \mathbf{R}_{\mathbf{xz}}^H &= \mathbf{W}_{m-1}^H \mathbf{R}_{\mathbf{xx}}^{\alpha H}, \end{aligned}$$

where an implicit lag parameter equal to the duration of one PCCA processing block is assumed for $\mathbf{R}_{\mathbf{xx}}^{\alpha}$. For the scalar case $z(k) \in \mathcal{C}^{\,1\times 1}$ the resulting weight update solutions are

$$c_{m,k} = \frac{\left(\mathbf{w}_{m-1}^H \mathbf{R}_{\mathbf{xx}}^{\alpha H}\right)}{\left(\mathbf{w}_{m-1}^H \mathbf{R}_{\mathbf{xx}}\mathbf{w}_{m-1}\right)}\mathbf{w}_{m,k-1}, \quad (24)$$
$$\mathbf{w}_{m,k} = \mathbf{R}_{\mathbf{xx}}^{-1}\mathbf{R}_{\mathbf{xx}}^{\alpha}\mathbf{w}_{m-1}c_{m,k}. \quad (25)$$

The Cyclic-WRPCCA is similarly implemented with the exception that the cyclic autocorrelation matrix $\mathbf{R}_{\mathbf{xx}}^{\alpha}$ is now computed using a lag parameter smaller than the PCCA processing block delay. For a single signal of interest, data recycling is incorporated to yield the Cyclic-WRPCCA/DR update,

$$c_{m,k} = \frac{\left(\mathbf{w}_{m,k-1}^H \mathbf{R}_{\mathbf{xx}}^{\alpha H}\right)}{\left(\mathbf{w}_{m,k-1}^H \mathbf{R}_{\mathbf{xx}}\mathbf{w}_{m,k-1}\right)}\mathbf{w}_{m,k-1}, \quad (26)$$
$$\mathbf{w}_{m,k} = \mathbf{R}_{\mathbf{xx}}^{-1}\mathbf{R}_{\mathbf{xx}}^{\alpha}\mathbf{w}_{m,k-1}c_{m,k}. \quad (27)$$

Notice that equation (27) is simply a scaled version of the previously discussed Phase-SCORE weight vector as determined by the basic power method. This is an interesting result providing some insight into the way in which the Phase-SCORE algorithm operates.

II.C CM-RPCCA

Exploitation of the constant modulus property can also be readily incorporated within the RPCCA framework. For a single signal of interest the necessary PCCA transformation is the complex limiting operation

$$z(k) = \frac{\mathbf{w}_{m-1}^H \hat{\mathbf{y}}(k)}{\left|\mathbf{w}_{m-1}^H \hat{\mathbf{y}}(k)\right|},$$

where the form of $\hat{\mathbf{y}}(k)$ is determined by the desired PCCA structure as follows,

1. *CM-RPCCA:* $\hat{\mathbf{y}}(k) = \mathbf{w}_{m-1}^H \mathbf{x}_{m-1}(k)$,
2. *CM-WRPCCA:* $\hat{\mathbf{y}}(k) = \mathbf{w}_{m-1}^H \mathbf{x}_m(k)$,
3. *CM-WRPCCA/DR:* $\hat{\mathbf{y}}(k) = \mathbf{w}_{m,j}^H \mathbf{x}_m(k)$.

For the CM-RPCCA and CM-WRPCCA techniques the iterative weight solutions are given by,

$$\begin{aligned} c_{m,k} &= R_{zz}^{-1}\mathbf{R}_{\mathbf{xz}}^H \mathbf{w}_{m,k-1}, && (28)\\ \mathbf{w}_{m,k} &= \mathbf{R}_{\mathbf{xx}}^{-1}\mathbf{R}_{\mathbf{xz}} c_{m,k}, && (29)\end{aligned}$$

with the first iteration initialized by $\mathbf{w}_{m,k-1} = \mathbf{w}_{m-1}$. The CM-WRPCCA/DR requires additional explicit recalculation of $z(k)$ at each iteration as follows,

$$\begin{aligned} z(k) &= \frac{\mathbf{w}_{m,j-1}^H \mathbf{x}_m(k)}{\left|\mathbf{w}_{m,j-1}^H \mathbf{x}_m(k)\right|}, && (30)\\ c_{m,j} &= R_{zz}^{-1}\mathbf{R}_{\mathbf{xz}}^H \mathbf{w}_{m,j-1}, && (31)\\ \mathbf{w}_{m,j} &= \mathbf{R}_{\mathbf{xx}}^{-1}\mathbf{R}_{\mathbf{xz}} c_{m,j}. && (32)\end{aligned}$$

III The Feedback PCCA

A natural extension of the RPCCA technique is to permit instantaneous feedback in place of the block delay operation. In doing so, the usual PCCA eigenequations no longer apply (in general) since the training signal set requires the instantaneous value of the beamformer weight $\mathbf{W}$. However, in certain applications the solution of the PCCA optimization criterion (modified to exclude the orthogonality constraint) can still be obtained either in closed form or in an iterative fashion. These new techniques are referred to as *Feedback PCCA* (FBPCCA).

III.A Cyclic-PCCA with Feedback

The PCCA feedback structure for exploitation of cyclostationarity is characterized by the transformation

$$\mathbf{f}\left[\mathbf{x}(\cdot), \mathbf{y}(\cdot)\right](k) = \mathbf{y}(k-n)e^{+j2\pi\alpha k}.$$

For the scalar case $z(k) \in \mathcal{C}^{1\times 1}$ the desired objective function is given by

$$\min_{\mathbf{W},c} \left\langle \left|\mathbf{w}^H\mathbf{x}(k) - c^*\mathbf{w}^H\mathbf{x}(k-n)e^{+j2\pi\alpha k}\right|^2\right\rangle. \quad (33)$$

Taking the gradient of (33) with respect to $\mathbf{w}$ and setting the result equal to zero we obtain

$$\mathbf{R}_{\mathbf{xx}}^{-1}\left[c\mathbf{R}_{\mathbf{xx}}^{\alpha}(n) + c^*\mathbf{R}_{\mathbf{xx}}^{\alpha H}(n)\right]\mathbf{w} = \left(|c|^2+1\right)\mathbf{w}. \quad (34)$$

Similarly, setting the derivative with respect to c equal to zero yields

$$c = \frac{\mathbf{w}^H\mathbf{R}_{\mathbf{xx}}^{\alpha H}(n)\mathbf{w}}{\mathbf{w}^H\mathbf{R}_{\mathbf{xx}}\mathbf{w}}. \quad (35)$$

While a closed form solution for equations (34) and (35) is not readily apparent, an alternating iterative approach can be applied as follows ,

$$\begin{aligned} c_{k+1} &= \frac{\mathbf{w}_k^H\mathbf{R}_{\mathbf{xx}}^{\alpha H}(n)\mathbf{w}_k}{\mathbf{w}_k^H\mathbf{R}_{\mathbf{xx}}\mathbf{w}_k},\\ \mathbf{w}_{k+1} &= \mathbf{R}_{\mathbf{xx}}^{-1}\left[c_{k+1}\mathbf{R}_{\mathbf{xx}}^{\alpha}(n) + c_{k+1}^*\mathbf{R}_{\mathbf{xx}}^{\alpha H}(n)\right]\mathbf{w}_k.\end{aligned}$$

Note that this approach yields a weight vector $\mathbf{w}$ that is the dominant mode of the eigenequation

$$\mathbf{R}_{\mathbf{xx}}^{-1}\left[c\mathbf{R}_{\mathbf{xx}}^{\alpha}(n) + c^*\mathbf{R}_{\mathbf{xx}}^{\alpha H}(n)\right]\mathbf{w} = \lambda\mathbf{w}, \quad (36)$$

in which $\lambda = \left(|c|^2+1\right)$ is the maximum eigenvalue. This implies that we are selecting the maximum value for $|c|$ subject to the constraint of (34). Note that without this constraint the quantity $|c|$ is maximized by the dominant eigenvector of $\mathbf{R}_{\mathbf{xx}}^{-1}\mathbf{R}_{\mathbf{xx}}^{\alpha H}(n)\mathbf{w} = \lambda\mathbf{w}$. Had we selected the alternative convention $\mathbf{z}(k) = \mathbf{y}(k-n)e^{-j2\pi\alpha k}$, (using the opposite sign for α) the result would have been equivalent to the Phase-SCORE weight vector.

III.B CM-PCCA with Feedback

Instantaneous feedback can be similarly incorporated within the CM-RPCCA structure. For single signal extraction the desired objective function is

$$\min_{\mathbf{W},c} \left\langle \left|\mathbf{w}^H\mathbf{x}(k) - c^*\frac{\mathbf{w}^H\mathbf{x}(k)}{|\mathbf{w}^H\mathbf{x}(k)|}\right|^2\right\rangle. \quad (37)$$

Taking the gradient of equation (37) with respect to $\mathbf{w}$ and setting the result equal to zero we obtain

$$\mathbf{w} = \mathrm{Re}\{c\}\mathbf{R}_{\mathbf{xx}}^{-1}\left\langle \frac{1}{2}(\mathbf{w}^H\mathbf{x}\mathbf{x}^H\mathbf{w})^{-1/2}\mathbf{x}\mathbf{x}^H\mathbf{w}\right\rangle, \quad (38)$$

where the time index k has been dropped for convenience. Similarly, taking the derivative of equation (37) with respect to c and setting the result equal to zero we obtain

$$c = \left\langle (\mathbf{w}^H\mathbf{x}\mathbf{x}^H\mathbf{w})^{1/2}\right\rangle. \quad (39)$$

Recall that $y = \mathbf{w}^H\mathbf{x}$ and that $(\mathbf{w}^H\mathbf{x}\mathbf{x}^H\mathbf{w})^{1/2} = (yy^*)^{1/2} = |y|$, therefore $\mathrm{Re}\{c\} = c$ and (38) can be reexpressed as

$$\mathbf{w} = c\mathbf{R}_{\mathbf{xx}}^{-1}\left\langle \frac{\mathbf{x}y^*}{|y|}\right\rangle. \quad (40)$$

Letting $z = y/|y|$ we obtain $\mathbf{w} = c\mathbf{R}_{\mathbf{xx}}^{-1}\mathbf{R}_{\mathbf{xz}}$. The complete iterative CM-FBPCCA algorithm is then given by

$$
\begin{aligned}
y_k &= \mathbf{w}_{k-1}^H \mathbf{x}, \\
c_k &= \langle |y_k| \rangle, \\
z_k &= \frac{y_k}{|y_k|}, \\
\mathbf{w}_k &= c_k \mathbf{R}_{\mathbf{xx}}^{-1}\mathbf{R}_{\mathbf{xz}}.
\end{aligned}
$$

Note that one iteration of the CM-FBPCCA algorithm is equivalent to the LS-CMA technique previously discussed, differing only by the real scale factor c_k.

IV Constrained RPCCA

In many applications it may be desirable to constrain the PCCA training signal weight matrix $\mathbf{C}$ to ensure that certain signal properties are included in (or excluded from) the training set. In this section, two approaches to this problem are considered. The first technique places a linear constraint on the training set composition using a type of Generalized Sidelobe Canceller (GSC) [6, 8] structure. The second approach is ad hoc in nature, and is designed to work with the iterative-power method implementations of the RPCCA processor.

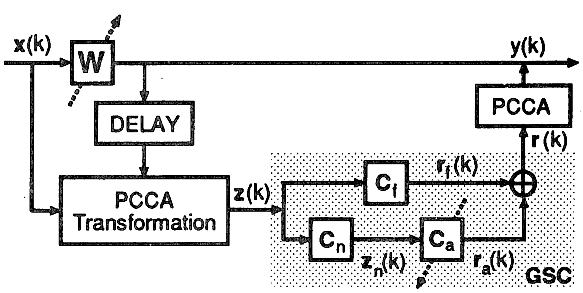

Figure 3: Recursive PCCA with Linearly Constrained Training Set

IV.A Linearly Constrained PCCA

Linear constraints on the training set composition can be realized by incorporating a processor in the form of a Generalized Sidelobe Canceller in the RPCCA training signal path as shown in Figure 3. Here the GSC will permit the PCCA to linearly constrain the composition of the training set $\mathbf{r}(k)$ with regard to the transformed data $\mathbf{z}(k)$. For $\mathbf{C} \in \mathcal{C}^{P\times L}$ a set of Q linear constraints is specified by the constraint equation

$$\mathbf{D}^H\mathbf{C} = \mathbf{F}, \tag{41}$$

where $\mathbf{D} = [\mathbf{d}_1\ \mathbf{d}_2 \cdots \mathbf{d}_Q]$ is the $P \times Q$ 'constraint matrix' and $\mathbf{F}$ is the $Q \times L$ matrix of constraint values or weights. The training signal $\mathbf{r}(k)$ is comprised of a 'fixed' and an 'adaptive' component given by

$$\mathbf{r}_f(k) = \mathbf{C}_f^H \mathbf{z}(k) \tag{42}$$

and

$$\mathbf{r}_a(k) = \mathbf{C}_a^H \mathbf{z}_n(k), \tag{43}$$

respectively, where $\mathbf{z}_n(k) = \mathbf{C}_n^H\mathbf{z}(k)$. Here the fixed weight $\mathbf{C}_f^H$ is a $P \times L$ matrix determined by the constraint matrix and constraint values to be

$$\mathbf{C}_f = \mathbf{D}\left(\mathbf{D}^H\mathbf{D}\right)^{-1}\mathbf{F}.$$

The 'nulling weight' $\mathbf{C}_n$ is a full-column-rank $P \times (P-Q)$ matrix with columns spanning the left null space of $\mathbf{D}$. The adaptive weight component $\mathbf{C}_a$ is a $(P-Q) \times L$ matrix that is adapted by the PCCA process. The overall effective weighting in the PCCA training signal path is then given by

$$\mathbf{C} = \mathbf{C}_f + \mathbf{C}_n\mathbf{C}_a,$$

with dimension $P \times L$.

The minimization criterion for the linearly constrained recursive PCCA (LC-RPCCA) is given by

$$\min_{\mathbf{W},\mathbf{C}_a} \left\langle \left| \mathbf{W}^H\mathbf{x}(k) - (\mathbf{C}_f + \mathbf{C}_n\mathbf{C}_a)^H \mathbf{z}(k) \right|^2 \right\rangle . \tag{44}$$

Taking the gradient of (44) with respect to $\mathbf{W}$ and setting the result equal to zero we obtain

$$\mathbf{W} = \mathbf{R}_{\mathbf{xx}}^{-1}\mathbf{R}_{\mathbf{xz}}\left(\mathbf{C}_f + \mathbf{C}_n\mathbf{C}_a\right). \tag{45}$$

Similarly, taking the gradient of (44) with respect to $\mathbf{C}_a$ and again setting the result equal to zero we obtain

$$
\begin{aligned}
\mathbf{C}_a &= \left[\mathbf{C}_n^H\mathbf{R}_{\mathbf{zz}}\mathbf{C}_n\right]^{-1}\left[\mathbf{C}_n^H\mathbf{R}_{\mathbf{xz}}^H\right]\mathbf{W} \\
&\quad - \left[\mathbf{C}_n^H\mathbf{R}_{\mathbf{zz}}\mathbf{C}_n\right]^{-1}\left[\mathbf{C}_n^H\mathbf{R}_{\mathbf{zz}}\mathbf{C}_f\right]. \qquad (46)
\end{aligned}
$$

The procedure for implementing the LC-RPCCA is then summarized as follows:

1. Specify the constraint equation $\mathbf{D}^H\mathbf{C} = \mathbf{F}$,
2. Select an arbitrary full-column-rank matrix $\mathbf{C}_n$ such that $\mathbf{D}^H\mathbf{C}_n = \mathbf{0}$,
3. Compute the fixed portion of the training set weight vector according to $\mathbf{C}_f = \mathbf{D}\left(\mathbf{D}^H\mathbf{D}\right)^{-1}\mathbf{F}$,
4. Initialize $\mathbf{W}$ with arbitrary orthogonal columns,

5. Compute $\mathbf{C}_a = \left[\mathbf{C}_n^H \mathbf{R}_{zz} \mathbf{C}_n\right]^{-1} \left[\mathbf{C}_n^H \mathbf{R}_{xz}^H\right] \mathbf{W} - \left[\mathbf{C}_n^H \mathbf{R}_{zz} \mathbf{C}_n\right]^{-1} \left[\mathbf{C}_n^H \mathbf{R}_{zz} \mathbf{C}_f\right]$,

6. Orthogonalize $\mathbf{C}_a$ using a Gram-Schmidt Orthogonalization,

7. Compute $\mathbf{W} = \mathbf{R}_{xx}^{-1} \mathbf{R}_{xz} \left(\mathbf{C}_f + \mathbf{C}_n \mathbf{C}_a\right)$,

8. Orthogonalize $\mathbf{W}$ using a Gram-Schmidt Orthogonalization,

9. Iterate steps (5) through (8) until convergence.

IV.B Magnitude Constrained PCCA

One potential drawback in using the LC-RPCCA technique is that the constraint conditions necessarily constrain both the magnitude and phase of the fixed portion of $\mathbf{C}$. In certain applications the optimal phase relationship between multiple training signals may not be known apriori, hence it would be advantageous to permit the PCCA processor to freely adapt the $\mathbf{C}$ weight phases while constraining the magnitudes. This can be accomplished with a straightforward complex limiting operation performed at each power method iteration. This new technique is referred to as the *magnitude constrained RPCCA* (MC-RPCCA) and is summarized as follows for the $L = 1$ scalar case:

1. Specify the constraint matrix $\mathbf{d} \in \mathcal{R}^{P \times 1}$ such that the desired $|\mathbf{c}| = [d_1\ d_2 \cdots d_P]^T$,

2. Select an arbitrary initial weight vector $\mathbf{w}$,

3. Compute $\mathbf{c}_k$ using one iteration of an *unconstrained* power method,

4. Apply the magnitude constraint to obtain $\hat{\mathbf{c}}_k = \left[\begin{array}{cccc} d_1 \frac{c_1}{|c_1|} & d_2 \frac{c_2}{|c_2|} & \cdots & d_P \frac{c_P}{|c_P|} \end{array}\right]^H$,

5. Complete the power method iteration for $\mathbf{w}_k$ using the constrained $\hat{\mathbf{c}}_k$,

6. Iterate steps (3) through (5) until convergence.

Note that the constraint magnitudes d_i need not be specified for all elements of the $\mathbf{c}$ weight. By omitting the complex limiting operation of step (4) for selected elements of $\mathbf{c}$, those elements may then be allowed to adapt freely - thereby remaining unconstrained.

V Simulation Results

Simulation performance results for the RPCCA, FBPCCA, and constrained PCCA are now presented. Three monte carlo experiments are conducted using a 5 element linear array geometry with 1/2 wavelength interelement spacing. Background noise is assumed to be Gaussian and spatially white. For simplicity, the analysis is limited to the single-signal extraction case with performance of the new techniques compared to that of established beamforming approaches.

V.A Cyclostationarity Results

Performance of the new cyclostationarity exploiting RPCCA techniques is evaluated for two signal environments. Techniques considered include Cross-SCORE, Phase-SCORE (or equivalently Cyclic-WRPCCA/DR), Cyclic-WRPCCA, and Cyclic-FBPCCA. The signal environments are described in Table 1.

Table 1: Cyclic-PCCA Environment

Environment 1			
Parameter	*SOI*	*SNOI1*	*SNOI2*
type	BPSK	-	-
SNR	20dB	-	-
AOA	10°	-	-
BW	0.25	-	-

Environment 2			
Parameter	*SOI*	*SNOI1*	*SNOI2*
type	BPSK	FM	FM
SNR	10dB	20dB	15dB
AOA	10°	50°	80°
BW	0.25	0.1	0.2

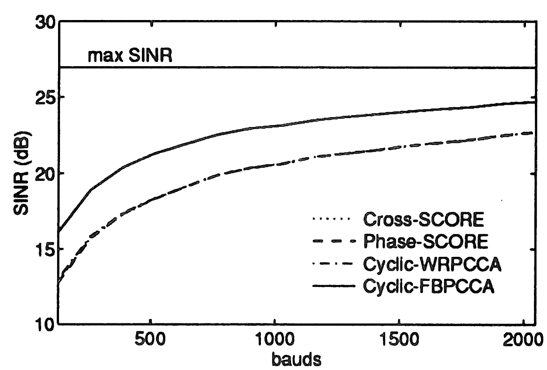

Figure 4: Environment 1 Cyclic-PCCA SINR.

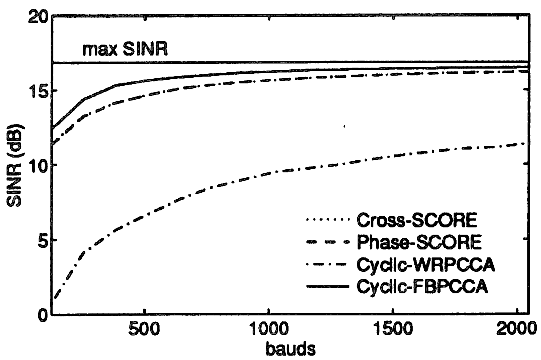

Figure 5: Environment 2 Cyclic-PCCA SINR.

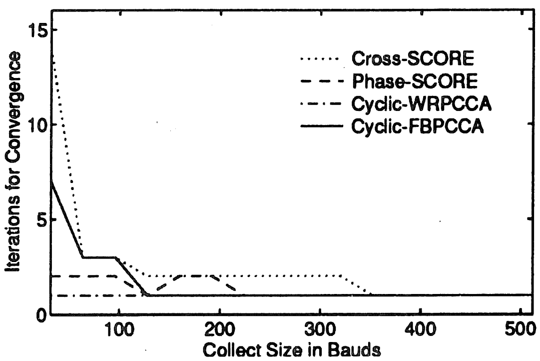

Figure 6: Environment 1 Cyclic-PCCA Convergence.

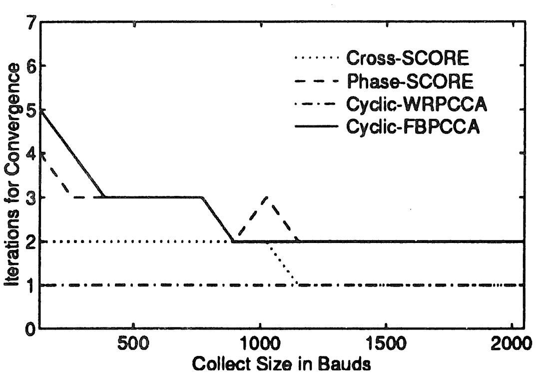

Figure 7: Environment 2 Cyclic-PCCA Convergence.

Output SINR performance results are presented in Figures 4 and 5 where the output SINR is evaluated as a function of the number of baud periods processed (collect time). Each algorithm is evaluated over 100 trials with the number of power method iterations fixed at five and a PCCA block size of 512 data points. Convergence performance is shown in Figures 6 and 7 and is characterized in terms of the number of power method iterations required to obtain an output SINR within 1% of the final output SINR. Results indicate the best overall performance is obtained by the FBPCCA for both signal environments. The number of iterations required for convergence appears comparable for all techniques.

V.B Constant Modulus Results

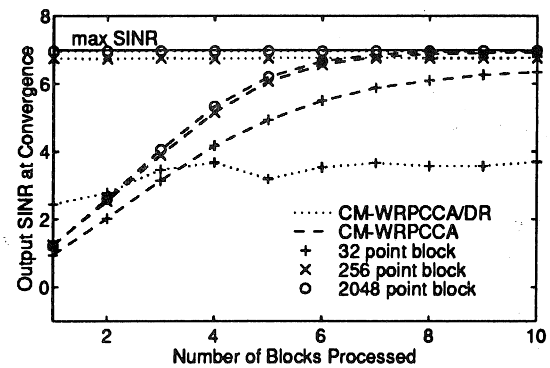

Figure 8: Environment 3 CM-PCCA SINR.

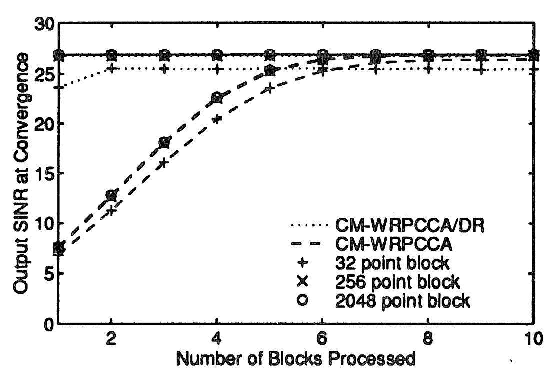

Figure 9: Environment 4 CM-PCCA SINR.

Many of the CM techniques discussed yield equivalent weight vectors (related by a real or complex scale factor) hence, a fairly complete performance characterization can be obtained by examining only the CM-WRPCCA/DR. Two signal environments are considered and are summarized in Table 2. The first performance test examines output SINR as a function of the number and size of PCCA data blocks

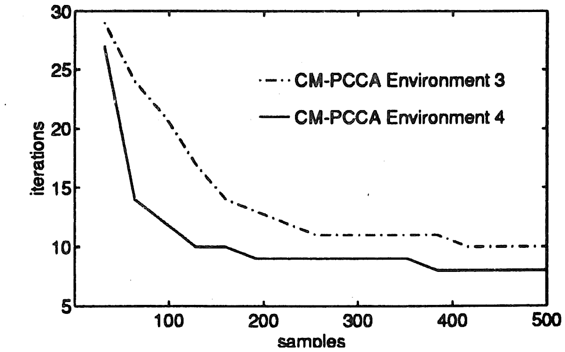

Figure 10: CM-PCCA Convergence.

processed. The number of power method iterations is fixed at 20. Results are presented in Figures 8 and 9 for block sizes of 32, 256, and 2048 data points. CM-WRPCCA (LS-CMA) results are obtained by examining the CM-WRPCCA/DR performance after one power method iteration.

Table 2: CM-PCCA Environment

Environment 3			
Parameter	*SOI*	*SNOI1*	*SNOI2*
type	FM	-	-
SNR	0dB	-	-
AOA	10°	-	-
BW	0.25	-	-

Environment 4			
Parameter	*SOI*	*SNOI1*	*SNOI2*
type	FM	BPSK	BPSK
SNR	20dB	10dB	15dB
AOA	10°	50°	80°
BW	0.25	0.1	0.2

Convergence behavior is characterized in terms of the number of power method iterations required to obtain output SINR performance within 1% of the final output SINR. Results for this test are shown in Figure 10 for PCCA block sizes ranging from 32 to 512 data points. Two key observations may be noted from these results. First, there is a very close relationship between 'iterations' in the CM-WRPCCA/DR technique and 'blocks processed' in the CM-WRPCCA approach. For example, if the CM-WRPCCA/DR technique is shown to require 5 iterations for convergence, then CM-WRPCCA will require 5 consecutive data blocks for convergence. It appears that the processors do not differentiate between 'new data' - such as that encountered in the block update of CM-WRPCCA - or 'recycled data' as applied iteratively in CM-WRPCCA/DR. A second observation is that upon convergence of both techniques, the CM-WRPCCA actually outperforms CM-WRPCCA/DR by a small margin (typically less than 1dB).

V.C Constrained PCCA Results

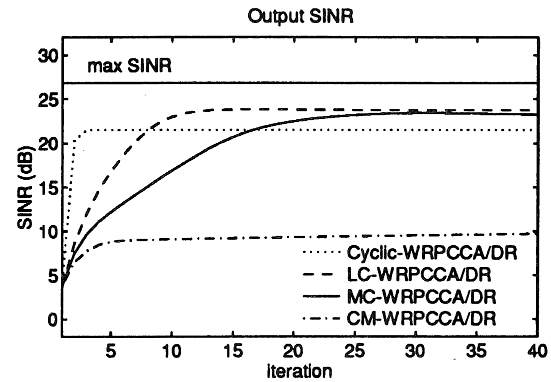

Figure 11: Constrained PCCA Scenario 1

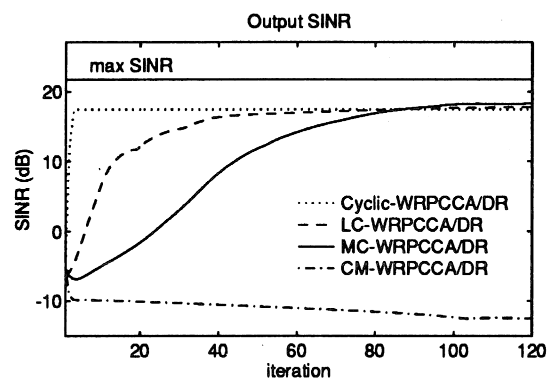

Figure 12: Constrained PCCA Scenario 2

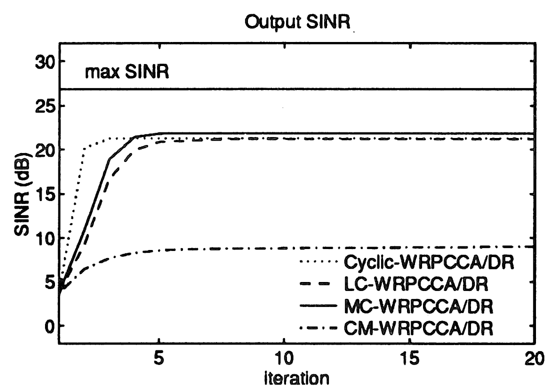

Figure 13: Constrained PCCA Scenario 3

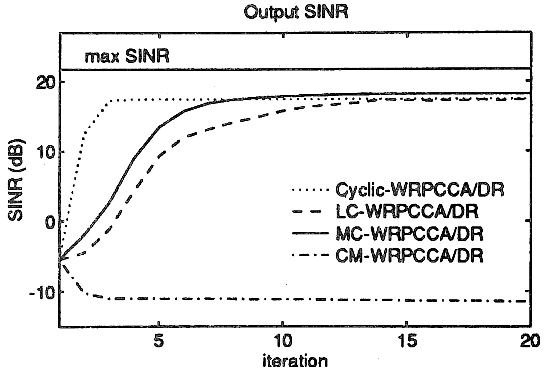

Figure 14: Constrained PCCA Scenario 4

Four criteria have been selected for quantifying constrained PCCA performance: 1) Percentage of Correct Captures, 2) Mean Output SINR, 3) Output SINR Standard Deviation, and 4) Iterations for Convergence. Two constrained PCCA techniques are considered in this evaluation - the linearly constrained WRPCCA/DR and the magnitude constrained WRPCCA/DR. Performance for these techniques is compared with the unconstrained Cyclic-WRPCCA/DR and CM-WRPCCA/DR approaches previously discussed.

The simulated signal environment is described in Table 3. Due to the use of rectangular pulse shaping in both BPSK formats, all three signals exhibit the constant modulus property.

Table 3: Constrained PCCA Signal Parameters.

signal	*modulation*	*AOA*	*SNR*	*BW*	*carrier*
1	BPSK	10°	20 dB	0.25	0.0
2	BPSK	50°	15 dB	0.1	0.0
3	FM	80°	10 dB	0.1	0.0

The training set is comprised of three training signals as follows,

$$\mathbf{z}_l(k) = \begin{bmatrix} \mathbf{w}_{m,l-1}^H \mathbf{x}_m(k-2) e^{+j2\pi(k-2)/4} \\ \mathbf{w}_{m,l-1}^H \mathbf{x}_m(k-5) e^{+j2\pi(k-5)/10} \\ \frac{\mathbf{w}_{m,l-1}^H \mathbf{x}_m(k)}{|\mathbf{w}_{m,l-1}^H \mathbf{x}_m(k)|} \end{bmatrix} \quad (47)$$

where each row of $\mathbf{z}_l(k)$ represents a unique training signal and l is the power method iteration index. Signals 1 and 2 exploit cyclostationarity while signal 3 exploits the constant modulus property.

Four constraint scenarios are considered and summarized in Table 4. The first two constraints place unity weighting on one of the cyclostationary training signals while allowing the others to freely adapt.

Table 4: Constraint Matrices for Constrained PCCA.

scenario	**D**	**F**	*Target*
1	$\begin{bmatrix} 1 & 0 & 0 \end{bmatrix}^T$	[1]	BPSK 1
2	$\begin{bmatrix} 0 & 1 & 0 \end{bmatrix}^T$	[1]	BPSK 2
3	$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}^T$	$\begin{bmatrix} 1 & 3 \end{bmatrix}^T$	BPSK 1
4	$\begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}^T$	$\begin{bmatrix} 1 & 3 \end{bmatrix}^T$	BPSK 2

Note that a value of zero in the **D** matrix implies a particular entry is to be unconstrained (not constrained to a value of zero). Constraint scenarios 3 and 4 utilize two constraints each - combining the cyclostationarity and modulus training signals. The modulus property is weighted slightly more than cyclostationarity, with a 3:1 weight ratio for their respective training signals.

The results of the constrained PCCA evaluation are presented in Figures 11-14 for the four constraint scenarios. All scenarios are evaluated over 100 monte carlo trials in which each technique operates on a single 4096 point data block using up to 120 power method iterations. A summary of quantitative results is provided in Tables 5 and 6.

In general, the constrained PCCA processors outperform the unconstrained processors, but at the expense of increased power method iterations required. The unconstrained CM-WRPCCA/DR performs poorly in all scenarios. The performance improvement for the constrained techniques is evidenced in the higher output SINRs obtained - up to 2.2dB higher in some scenarios. In most cases the MC-WRPCCA/DR performance is superior to that of LC-WRPCCA/DR. Note that the relatively poor 'correct capture' performance observed in scenario 2 appears to be due to the lack of a constraint on the constant modulus weight which tends to grow very large. This results in the modulus property being weighted much more strongly than the cyclostationarity property is, causing the processor to switch capture from signal 2 to signal 1 - the strongest constant modulus signal in the environment. Finally, the variance of the output SINR appears comparable for both the constrained and unconstrained approaches.

Table 5: Constrained PCCA Performance I.

	% correct captures		*mean output SINR (dB)*		
scenario	LC	MC	LC	MC	cyclic
1	59	100	23.798	23.28	21.53
2	36	81	17.76	18.35	17.43
3	100	100	21.21	21.81	21.26
4	62	100	17.37	18.16	17.44

Table 6: Constrained PCCA Performance II.

	σ output SINR (dB)			*iterations for convergence*		
scenario	LC	MC	cyclic	LC	MC	cyclic
1	1.25	2.55	1.58	13	30	3
2	3.74	3.33	1.37	60	100	3
3	2.05	1.42	1.86	6	5	3
4	2.37	1.26	1.46	14	11	3

VI Conclusions

In this paper the Programmable Canonical Correlation Analyzer (PCCA) has been modified to include recursion, feedback and training-set constraints. Development has included the establishment of appropriate optimization criteria, optimal weight solutions (when tractable), and efficient power method implementations. The resulting family of adaptive spatial filters has been shown to include previously established beamforming techniques, as well as several new processors. The performance of these techniques has been evaluated via monte carlo simulation and characterized in terms of output SINR and convergence behavior. In most cases, the new PCCA techniques are able to outperform established techniques, but at the expense of increased computational burden associated with the higher number of iterations required for convergence.

Appendix A - Power Methods

Basic Power Method

Given the eigenequation $\mathbf{A}\mathbf{w} = \lambda\mathbf{w}$ the dominant (or maximum) eigenvector can be iteratively estimated by the basic power method

$$\hat{\mathbf{w}}_{k+1} = \gamma_k \mathbf{A}\hat{\mathbf{w}}_k,$$

where γ_k is a normalization constant typically chosen to be the inverse of the first element of the $\hat{\mathbf{w}}_k$ vector. The basic power method converges to a solution that is a scalar multiple of the dominant eigenvector of $\mathbf{A}$.

Block Power Method [12]

The block power method extends the basic power method to include estimation of multiple eignevectors as follows,

i. Initialize $\hat{\mathbf{W}}_k = [\hat{\mathbf{w}}_1\ \hat{\mathbf{w}}_2 \cdots \hat{\mathbf{w}}_p]$ such that the $\hat{\mathbf{w}}_i$ are orthonormal,

ii. $\hat{\mathbf{W}}_{k+1} = \mathbf{A}\hat{\mathbf{W}}_k$,

iii. Apply Gram-Schmidt Orthogonalization to $\hat{\mathbf{W}}_{k+1}$.

iv. Iterate steps (ii) and (iii).

Alternating Power Method [12]

A modified version of the power method can be applied in cases where two weight vectors are to be jointly determined. For the single-signal case, the system of eigenequations is represented by,

$$\begin{aligned} \left[\mathbf{Ab}D\mathbf{g}^H\right]\mathbf{w} &= \lambda\mathbf{w} \\ \left[D\mathbf{g}^H\mathbf{Ab}\right]c &= \lambda c, \end{aligned}$$

where $\mathbf{w}$ and c are the desired weight vectors, $\mathbf{A}$ is $M \times M$, D is scalar, and $\mathbf{b}$ and $\mathbf{g}$ are $M \times 1$ vectors. The dominant eigenvectors for this system are estimated by the recursion

$$\begin{aligned} c_k &= \gamma_k \left[\mathbf{D}\mathbf{g}^H\right]\mathbf{w}_{k-1} \\ \mathbf{w}_k &= \beta_k \left[\mathbf{Ab}\right] c_k, \end{aligned}$$

where γ_k and β_k are the appropriate normalization coefficients.

Alternating Block Power Method

The block power method and alternating power method can be combined to yield an alternating block power method for jointly determining multiple eigenvectors. The system of eigenequations is given by

$$\begin{aligned} \left[\mathbf{ABDG}^H\right]\mathbf{W} &= \mathbf{W}\Lambda \\ \left[\mathbf{DG}^H\mathbf{AB}\right]\mathbf{C} &= \mathbf{C}\Lambda, \end{aligned}$$

where $\mathbf{A}$ and $\mathbf{D}$ are $M \times M$ and $L \times L$ matrices, respectively, and $\mathbf{B}$ and $\mathbf{G}$ are $M \times L$ matrices. Representing the weight matrices by

$$\mathbf{C} = [\mathbf{c}_1\ \mathbf{c}_2 \cdots \mathbf{c}_L]$$

$$\mathbf{W} = [\mathbf{w}_1\ \mathbf{w}_2 \cdots \mathbf{w}_L],$$

the technique is as follows:

i. Initialize $\hat{\mathbf{W}}_k = [\hat{\mathbf{w}}_1 \ \hat{\mathbf{w}}_2 \cdots \hat{\mathbf{w}}_L]$ such that the $\hat{\mathbf{w}}_i$ are orthonormal,

ii. Compute $\hat{\mathbf{C}}_{k+1} = [\mathbf{D}\mathbf{G}^H]\,\hat{\mathbf{W}}_k$,

iii. Apply the Gram-Schmidt Orthogonalization Procedure to $\hat{\mathbf{C}}_{k+1}$ to obtain $\hat{\hat{\mathbf{C}}}_{k+1}$.

iv. Compute $\hat{\mathbf{W}}_{k+1} = [\mathbf{A}\mathbf{B}^H]\,\hat{\hat{\mathbf{C}}}_{k+1}$,

v. Apply the Gram-Schmidt Orthogonalization Procedure to $\hat{\mathbf{W}}_{k+1}$,

vi. Iterate steps (ii) through (v).

Acknowledgements

The authors wish to thank Rome Laboratories IRAA and E-Systems Greenville Division for supporting this work and its publication.

References

[1] B. G. Agee. The least-squares CMA: a new technique for rapid correction of constant modulus signals. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 953–956, April 1986.

[2] B. G. Agee, S. V. Schell, and W. A. Gardner. Self-coherence restoral: A new approach to blind adaptation of antenna arrays. In *Proceedings of the Twenty First Asilomar Conference on Signals, Systems, and Computers*, pages 589–593, 1987.

[3] T. W. Anderson. *An Introduction to Multivariate Statistical Analysis.* John Wiley and Sons, New York, 1984.

[4] W. A. Gardner, J. L. Schenck, and S. V. Schell. Programmable blind adaptive spatial filtering. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, Adalaide, South Australia, April 1994.

[5] G. H. Golub and C. F. Van Loan. *Matrix Computations, 2nd ed.* Johns Hopkins University Press, Baltimore, MD, 1989.

[6] L. J. Griffiths and C. W. Jim. An alternative approach to linearly constrained adaptive beamforming. *IEEE Transactions on Antennas and Propagation*, AP-30:27–34, January 1982.

[7] R. A. Johnson and D. W. Wichern. *Applied Multivariate Statistical Analysis.* Prentice Hall, Englewood Cliffs, New Jersey, 1992.

[8] M. Rude and L. Griffiths. Incorporation of linear constraints into the constant modulus algorithm. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 968–971, February 1989.

[9] S. Schell and B. G. Agee. Application of the SCORE algorithm and SCORE extensions to sorting in the rank-L spectral self-coherence environment. In *Proceedings of the Twenty Second Asilomar Conference on Signals, Systems, and Computers*, pages 274–278, 1988.

[10] S. V. Schell. An overview of sensor array processing for cyclostationary signals. In W. A. Gardner, editor, *Cyclostationarity in Communications and Signal Processing.* IEEE Press, New York, 1994.

[11] S. V. Schell and W. A. Gardner. Spatio-temporal filtering and equalization for cyclostationary signals. In C. T. Leondes, editor, *Digital Signal Processing Techniques and Applications.* Academic Press, New York, 1994.

[12] G. Strang. *Linear Algebra and its Applications.* Harcourt Brace Jovanovich College Publishers, Fort Worth, Texas, 1988.