

# Median Cascaded Cancellor for Robust Adaptive Array Processing

MICHAEL L. PICCIOLO, Senior Member, IEEE

KARL GERLACH, Fellow, IEEE  
Naval Research Laboratory

**A median cascaded canceller (MCC) is introduced as a robust multichannel adaptive array processor. Compared with sample matrix inversion (SMI) methods, it is shown to significantly reduce the deleterious effects of impulsive noise spikes (outliers) on convergence performance of metrics such as (normalized) output residue power and signal to interference-plus-noise ratio (SINR). For the case of no outliers, the MCC convergence performance remains commensurate with SMI methods for several practical interference scenarios. It is shown that the MCC offers natural protection against desired signal (target) cancellation when weight training data contains strong target components. In addition, results are shown for a high-fidelity, simulated, barrage jamming and nonhomogenous clutter environment. Here the MCC is used in a space-time adaptive processing (STAP) configuration for airborne radar interference mitigation. Results indicate the MCC produces a marked SINR performance improvement over SMI methods.**

Manuscript received October 18, 2001; revised October 22, 2002; released for publication April 10, 2003.

IEEE Log No. T-AES/39/3/818489.

Refereeing of this contribution was handled by P. Lombardo.

This work supported by the Office of Naval Research.

Authors' current addresses: M. Picciolo, SAIC, 4501 Daly Dr., Suite 400, Chantilly, VA 20151-3707, E-mail: (Michael.L.Picciolo@saic.com); K. Gerlach, Naval Research Laboratory, Code 5341, 4555 Overlook Ave. S.W., Washington, D.C. 20375.

U.S. Government work not protected by U.S. copyright.

0018-9251/03/\$17.00 2003 IEEE

## I. INTRODUCTION

Adaptive processors utilize multichannel (multi-sensor) measured input data to estimate the signal environment in order to mitigate interference and noise while maximizing the desired signal energy at the processor output. The ability to adapt is often required because adequate a priori knowledge of the interference and noise statistics is not available. For radar applications, the interference and noise environment often refers to, respectively, external interference that is correlated across input channels such as from jamming and ground clutter, and internal receiver thermal noise (denoted as "noise") that is present on each input channel but uncorrelated from channel to channel. To form weights used in a linear adaptive processor, the interference-plus-noise covariance matrix is often estimated, either directly or indirectly, using measured data from the input channels. These methods are referred to here as sample matrix inversion (SMI) [1] methods. It is generally assumed that jamming, clutter, and noise signals are independent from one another and additive in nature, and that noise is present in each input channel.

The signal to interference-plus-noise ratio (SINR) convergence measure of effectiveness (MOE) is the amount of data—the number of statistically independent and identically distributed (IID) samples—that is required so that the average SINR of the adaptive processor is nominally within 3 dB of optimum. Minimizing the convergence MOE (i.e., maximizing the convergence rate) is important for satisfactory performance in many practical situations, especially when the interference and noise environment is not strictly wide sense stationary. This may be due to, for example, nonhomogeneous clutter and/or a volatile jamming environment. In addition, computational complexity and cost generally increase when processing more samples.

The convergence MOE of the standard open loop (i.e., no feedback) adaptive linear technique, the SMI algorithm [1] (or any numerically equivalent implementation), can be attained using approximately  $2N$  IID samples per input channel for weight estimation in pure stationary Gaussian interference and noise environments. The integer  $N$  denotes the number of degrees of freedom (DOF) or number of input channels to the adaptive processor. For example,  $N$  is equal to the number of antenna channels (i.e., antenna elements or subarrays) for a spatially adaptive array processor, and equal to the number of space and time channels for a space-time adaptive processing (STAP) processor. This  $2N$  convergence MOE is shown in [1] to be independent of any external interference covariance matrix when the Gaussian assumption is strictly satisfied. Referred to as the SMI convergence MOE, it has become a benchmark used to assess convergence rates of adaptive processors.

In this work a new adaptive processing algorithm is introduced that has fast convergence performance commensurate with SMI methods for a limited number of benign interference and noise environments. A benign environment refers to pure Gaussian, (wide sense) stationary and outlier-free received data. However, the new method often has performance superior to SMI methods in nonbenign environments which are frequently encountered in real world scenarios [21]. This superior performance improvement in nonbenign interference and noise is due to its robustness against outliers and targets in the data used for weight training (training data), and due to its robustness to nonstationary data.

Reduced rank adaptive processing methods exist that converge faster than SMI methods, for example [8, 13]. However, the new algorithm as described here is considered to be a full rank processor, and it is compared here with (full rank) SMI methods only. Improving the robustness of reduced rank processors is a separate topic of research, e.g. [17].

Measured data can be “contaminated” with outlier values (impulsive noise spikes) due to several sources. For example, radar receivers may sense sidelobe clutter discretely [3], blinking jammers, and target returns (especially for a high pulse repetition frequency (HPRF) radar where near-in (friendly) targets can return strong signals). Additionally, outliers may be caused by general electromagnetic interference (EMI), cross-channel interference, intermodulation spikes, digital radio-frequency memory (DRFM) jammers, cover pulse electronic counter measures (ECM), intermittent bad data channels, range distributed targets in a high range resolution radar, and nearby in-band pulsed radars. These and other sources can corrupt the received data which then affects the estimate of the unknown interference-plus-noise covariance matrix.

The performance degradation of the SMI algorithm in the presence of outlier contaminant data can be attributed to the highly sensitive nature of covariance matrix estimates to even small amounts of impulsive noise. This notion is plainly supported by Huber [12, p. 199]: “Unfortunately, sample covariance matrices are excessively sensitive to outliers.” Results in Section IV illustrate how the convergence rates for SMI methods are especially sensitive to a specific target-like form of outlier that may contaminate the input data of the adaptive processor.

The initial treatment here is concerned with spatially adaptive array radars, followed by a STAP application pertaining to airborne radars. However, the median cascaded canceller (MCC) processor has general application in fields using adaptive signal processing and related techniques, for example, radar, sonar, acoustics, seismic array processing, image processing, medical imaging, hyperspectral imaging and spectrum estimation. The outline of

this work is as follows. In Section II we discuss how an SMI adaptive array processor may be equivalently implemented using a Gram-Schmidt cascaded canceller. In Section III an MCC structure is introduced as a robust surrogate for a Gram-Schmidt cascaded canceller. Theoretical convergence results are derived and compared with SMI methods for pure Gaussian interference and noise. In Section IV, STAP simulation results are shown, illustrating the robust convergence performance of the MCC processor compared with the SMI processor for a representative practical interference and noise environment for airborne radar.

## II. PRELIMINARIES

A general linear adaptive array processor with multiple linear constraints is referred to as a linearly constrained minimum variance (LCMV) beamformer or processor [4]. For the case of a single mainbeam constraint having unity gain, it is also known as a minimum variance distortionless response (MVDR) processor [4]. We refer to the generally complex-valued  $N \times 1$  MVDR optimal weight vector,  $\mathbf{w}_{\text{MVDR}} = \beta \mathbf{R}^{-1} \mathbf{s}$ , as the direct form MVDR solution, where  $\mathbf{R} = E\{\mathbf{q}\mathbf{q}^H\}$  is the  $N \times N$  interference and noise covariance matrix,  $\mathbf{q}$  is the  $N \times 1$  input channel vector,  $\mathbf{s}$  is the  $N \times 1$  normalized (unit norm) desired signal (target) steering vector, and the scalar  $\beta = 1/(\mathbf{s}^H \mathbf{R}^{-1} \mathbf{s})$  satisfies the unity gain constraint. The symbol  $E$  denotes expectation, and the superscript  $H$  denotes conjugate-transpose (Hermitian) operation. In adaptive applications,  $\mathbf{s}$  usually is assumed known or measured to sufficient accuracy, and  $\mathbf{R}$  is estimated adaptively from training data that usually is assumed to contain no target components. In this work vectors are denoted as bold face, lower case quantities; matrices are denoted as bold face, upper case quantities; and scalars are denoted as nonbold face, italic quantities.

Both LCMV and MVDR direct form processors have corresponding numerically equivalent canceller forms via the generalized sidelobe canceller methodology [4]. For brevity, only the MVDR processor and equivalent implementations are considered here, but results apply to the more general LCMV processor as well.

### A. General Linear Adaptive Array: Canceller Form

The canceller form of the MVDR processor consists of a deterministic, nonsingular,  $N \times N$  matrix transformation  $\mathbf{A}$  applied to the input vector  $\mathbf{q}$ , followed by a canceller adaptive processor as shown in Fig. 1 [4–6]. It is assumed, without loss of generality, that  $\mathbf{q}$  and therefore  $\mathbf{u}$  have zero-valued  $N \times 1$  mean vectors.

The  $N \times 1$  vector  $\mathbf{u} = \mathbf{A}\mathbf{q}$  resulting from the linear transformation has all desired signal energy

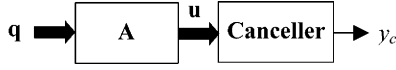


Fig. 1. MVDR processor shown in canceller form. Vector  $\mathbf{q}$  is input data vector,  $y_c$  is canceller scalar output,  $\mathbf{A}$  is nonsingular matrix transformation,  $\mathbf{u} = \mathbf{A}\mathbf{q}$  is canceller form of input data vector  $\mathbf{q}$ .

within  $\mathbf{q}$  placed into the first or “main” channel of  $\mathbf{u}$  (as well as transformed interference and noise), leaving only transformed interference and noise in the remaining “auxiliary” channels of  $\mathbf{u}$ . (One method to determine  $\mathbf{A}$  for the MVDR case is discussed in Section IVB.) The main channel of  $\mathbf{u}$  is labeled  $u_m = u_1$ , and auxiliary channels  $2, \dots, N$  are labeled  $\mathbf{u}_a = [u_2, \dots, u_N]^T$  where superscript T denotes transpose. The data vectors  $\mathbf{q}$  and  $\mathbf{u}$  as well as the output scalar  $y_c$  are understood to be sampled discretely in time; i.e.,  $\mathbf{q} = \mathbf{q}(k)$ ,  $\mathbf{u} = \mathbf{u}(k)$ , and  $y_c = y_c(k)$ , where  $k$  is discrete time index. In radar applications, samples in range usually directly correspond to samples in time and are substituted accordingly.

A canceller is a specific form of an adaptive array processor and is often referred to as a sidelobe canceller. The  $N \times 1$  normalized desired signal steering vector of the canceller is defined to be  $\mathbf{s} = \mathbf{1}_0$ , where  $\mathbf{1}_0 = [1 \ 0 \ 0 \ \dots \ 0]^T$ . The main channel is weighed by a fixed value of unity, and the auxiliary channels are adaptively weighted and subtracted from the main channel. The  $(N - 1) \times 1$  auxiliary channel optimal weight vector,  $\mathbf{w}_{a\text{opt}}$ , solves the Wiener-Hopf matrix equation

$$\mathbf{w}_{a\text{opt}} = \mathbf{R}_a^{-1} \mathbf{r}_{am}. \quad (1)$$

In (1),  $\mathbf{R}_a = E\{\mathbf{u}_a \mathbf{u}_a^H\}$  is the  $(N - 1) \times (N - 1)$  auxiliary channel covariance matrix, and  $\mathbf{r}_{am} = E\{\mathbf{u}_a u_m^*\}$  is the  $(N - 1) \times 1$  cross-correlation vector between the auxiliary channels and the main channel, where the symbol  $*$  denotes complex-conjugate.

Adaptive methods use a finite number of samples  $K$  to form an estimate of the optimum weight vector. For a canceller, the optimum auxiliary channel weight vector estimate,  $\hat{\mathbf{w}}_{a\text{opt}} = \hat{\mathbf{R}}_a^{-1} \hat{\mathbf{r}}_{am}$  ( $\hat{\cdot}$  denotes estimate), is found using the method of least squares [4] and yields estimates of the quantities in (1) as

$$\hat{\mathbf{R}}_a = \frac{1}{K} \sum_{k=1}^K \mathbf{u}_a(k) \mathbf{u}_a^H(k) \quad (2)$$

and

$$\hat{\mathbf{r}}_{am} = \frac{1}{K} \sum_{k=1}^K \mathbf{u}_a(k) u_m^*(k) \quad (3)$$

where the factors  $1/K$  in (2) and (3) are included for convenience but cancel each other in  $\hat{\mathbf{w}}_{a\text{opt}}$ . It is well known that, for Gaussian statistics, (2) and (3) are also maximum likelihood estimates of the covariance matrix and cross-correlation vector, respectively.

In this paper, the quantities  $\mathbf{q}(k)$ ,  $\mathbf{u}_a(k)$  and  $u_m(k)$ ,  $k = 1, \dots, K$ , are considered nonconcurrent weight training data; i.e., weight vectors derived from this data are not applied to this data; instead they are applied to independent data in order to mitigate interference and detect targets. (An exception to this practice is made in Section IV B to illustrate some performance results.) Also, assuming ergodicity, in the limit as  $K \rightarrow \infty$ , it can be shown that the right hand sides of (2) and (3) approach the corresponding quantities in (1).

As an alternative to least squares methodology, it can be shown that the SMI algorithm, as applied to the canceller input data  $\mathbf{u}$ , directly substitutes the maximum likelihood quantities (2) and (3) into the corresponding quantities in (1). Thus, the least squares method and the SMI method produce identical weight vectors, and we refer to these methods interchangeably.

The scalar output of the canceller form of an adaptive MVDR processor using the SMI algorithm is given explicitly as

$$y_c = \hat{\mathbf{w}}_{c\text{opt}}^H \mathbf{u} = \begin{bmatrix} 1 \\ -\hat{\mathbf{w}}_{a\text{opt}} \end{bmatrix}^H \mathbf{u} \quad (4)$$

where  $\hat{\mathbf{w}}_{c\text{opt}} = [1 \ -\hat{\mathbf{w}}_{a\text{opt}}]^T$  is the canceller's  $N \times 1$  optimal weight vector estimate. The direct form adaptive MVDR processor using the SMI algorithm substitutes the assumed desired signal steering vector  $\mathbf{s}$  and the maximum likelihood estimate of  $\mathbf{R}$ , i.e.,  $\hat{\mathbf{R}} = (1/K) \sum_{k=1}^K \mathbf{q}(k) \mathbf{q}^H(k)$ , into the corresponding quantities in  $\mathbf{w}_{\text{MVDR}} = \beta \mathbf{R}^{-1} \mathbf{s}$ , yielding  $\hat{\mathbf{w}}_{\text{MVDR}}$ . It can be shown that the direct form output,  $y_{\text{MVDR}} = \hat{\mathbf{w}}_{\text{MVDR}}^H \mathbf{q}$ , is numerically identical to the canceller form output  $y_c$  in (4).

## B. Cancellor: Gram-Schmidt Cascaded Cancellor Form

It is known that the canceller given by (4) has an equivalent Gram-Schmidt cascaded canceller (GSCC) form with a numerically identical output, both in the “steady state” [9] and in the “transient state” [5]. Transient state refers to the case where adaptive weights are estimated using a finite number of samples  $K$ , and steady state refers to the case when  $K \rightarrow \infty$  (optimally weighted). A method to derive the set of equivalent linear weights (4) from the cascaded set within the GSCC is described in [18].

If the GSCC is substituted for the canceller in Fig. 1, the result is a GSCC canceller form of MVDR processor. We use the phrase MVDR processor here to refer to any of the numerically equivalent MDVR forms: direct, canceller, or GSCC canceller. It can be shown that the three forms have identical outputs for optimally weighted as well as for adaptively weighted cases when using the SMI method. For the adaptive case we refer to these equivalent forms as SMI processors or methods.

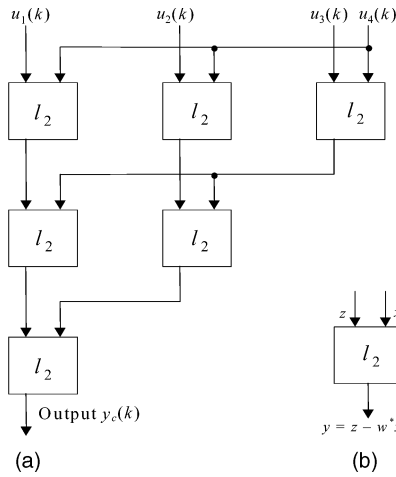


Fig. 2. (a) GSCC for  $N = 4$  case. (b) Any single building block canceller.

A GSCC is shown in Fig. 2(a) for the  $N = 4$  input channel case. It is comprised of six identical two-input canceller building blocks; any single block is represented in Fig. 2(b). The symbol  $l_2$  in each block indicates the weight estimate for that block is calculated by minimizing the square of an  $l_2$  vector norm, as will be discussed.

In the following, for notational simplicity, the left input (or local main channel) of any single building block is labeled  $z$ , the right input (or local auxiliary channel) is labeled  $x$ , and the output (or local output residue) is labeled  $y$  as shown in Fig. 2(b). Derived from mean square error (MSE) criterion, the optimal weight for each block is determined from (1) to be  $w_{\text{opt}} = R_{xx}^{-1}r_{xz}$ , where  $R_{xx}$  and  $r_{xz}$  denote the scalar auxiliary channel variance (power) and the scalar cross-correlation between the main and auxiliary channels, respectively. To produce the output  $y$ , each block subtracts from  $z$  the component of  $z$  that is correlated with  $x$ . In adaptive applications this process is approximated by choosing a weight such that the residual output  $y$  is numerically uncorrelated with  $x$  over a finite number of  $K$  values, where it is understood that  $y = y(k)$ ,  $z = z(k)$ , and  $x = x(k)$ , for  $k = 1, \dots, K$ . To do this the least squares method is used to minimize, over the infinite set of complex scalar weights  $w$ , the square of the  $l_2$  norm of the output vector  $y = z - w^*x$ . Specifically, for any single building block the optimal weight estimate is found as

$$\hat{w}_{\text{opt}} = \arg \min_w \left[ \sum_{k=1}^K |z(k) - w^*x(k)|^2 \right] \quad (5)$$

where,  $|\cdot|$  denotes magnitude. This results in the scalar adaptive Wiener-Hopf equation

$$\hat{w}_{\text{opt}} = \hat{R}_{xx}^{-1} \hat{r}_{xz} \quad (6)$$

where

$$\hat{R}_{xx} = \frac{1}{K} \sum_{k=1}^K |x(k)|^2 \quad (7)$$

and

$$\hat{r}_{xz} = \frac{1}{K} \sum_{k=1}^K x(k)z^*(k) \quad (8)$$

where the factors  $1/K$  in (7) and (8) are included for convenience but cancel each other in (6). Equations (7) and (8) are known also to be maximum likelihood estimates of the scalar covariance and cross-correlation, respectively, for Gaussian statistics. It is noted that (7) and (8) (and, more generally, (2) and (3)) are sensitive to outlier contaminants because even a single outlier in  $x(k)$  or  $z(k)$ ,  $k = 1, \dots, K$  may unboundedly bias the summations.

### III. MEDIAN CASCADED CANCELLER

It was discussed in Section II how a general adaptive array processor (LCMV or MVDR) can be transformed equivalently into its GSCC canceller form, which uses operationally identical, two-input canceller building blocks. With this in mind we modify the adaptive weight calculation algorithm for these  $l_2$  blocks in order to improve robustness to outliers that may be present in the data used for weight training. The linearly weighted form for each block,  $y = z - w^*x$ , is retained. However, the complex weight  $w$  is chosen using a new, robust, two-input canceller building block algorithm. In the limit as  $K \rightarrow \infty$ , the robust weight will be shown to converge to the same optimal weight as the Gram-Schmidt  $l_2$  block for a large and applicable class of interference and noise distributions. For this novel two-input building block, the nonlinear sample median function is used, and the new building block is subsequently labeled  $l_{\text{med}}$ .

#### A. Two-Input Median Cascaded Canceller: The $l_{\text{med}}$ Building Block

To derive the two-input  $l_{\text{med}}$  adaptive weight algorithm, consider the solution of (6) (which uses an  $l_2$  cost function) for the case of a single point average (i.e.,  $K = 1$ ). The solution is seen by inspection to be  $\hat{w}_{\text{opt}_1} = [z(1)/x(1)]^*$ , where the subscript 1 denotes a single sample solution. (This remains the single sample solution even for cost functions having exponents other than two, i.e., for any real, positive exponent  $p$ , describing any general  $l_p$  cost function.) It is noted that the quantity  $\hat{w}_{\text{opt}_1}$  is a random variable for different values of  $k$ , and may be denoted as  $\hat{w}_{\text{opt}_1}(k)$ . By itself,  $\hat{w}_{\text{opt}_1}(k)$  zeros the output of a two-input canceller if calculated

and applied at each time instant  $k$  and thus has no value.

However, we show that the complex median (and mean) of  $\hat{w}_{\text{opt}_1}(k)$  equals the optimal weight  $w_{\text{opt}} = R_{xx}^{-1}r_{xz}$  under certain broad assumptions to be discussed shortly. Complex median refers to the combination of the medians of the real and imaginary components of a random variable as determined separately. For adaptive applications, it is shown that as  $K \rightarrow \infty$  the sample complex median of  $\hat{w}_{\text{opt}_1}(k)$ ,  $k = 1, \dots, K$ , converges to  $w_{\text{opt}}$  in the mean square sense.

For this analysis, it is assumed that both  $z$  and  $x$  each have a zero-mean, Gaussian probability density function (pdf) and associated cumulative distribution function (cdf) for each of their real and imaginary parts. (We point out that the asymptotic results to be shown also hold under the broad assumptions that the input pdfs are symmetric, i.e., if  $z$  and  $x$  each have complex, symmetric densities).

The adaptive weight for the two-input  $l_{\text{med}}$  algorithm is calculated as follows. Form the set  $\omega(k) = \hat{w}_{\text{opt}_1}(k) = [z(k)/x(k)]^*$ ,  $k = 1, 2, \dots, K$ , and assume  $K$  is odd for simplicity. Take the sample median (denoted by MED in capital letters) of the real parts of  $\{\omega(k)\}$  as the real part of the new optimal weight estimate. Next, take the sample median of the imaginary parts of  $\{\omega(k)\}$  as the imaginary part of the new optimal weight estimate. It is shown that, with no outliers and as  $K \rightarrow \infty$ , the resulting sample complex median weight,

$$w_{\text{MED}} = \text{MED}_{k=1 \text{ to } K} \left[ \text{Re} \left( \frac{z^*(k)}{x^*(k)} \right) \right] + j \left\{ \text{MED}_{k=1 \text{ to } K} \left[ \text{Im} \left( \frac{z^*(k)}{x^*(k)} \right) \right] \right\} \quad (9)$$

converges asymptotically to  $w_{\text{opt}}$  like an  $l_2$  block weight using the same input data. In (9),  $j = \sqrt{-1}$  is the unit imaginary number. For  $K$  even, an adequate definition of the sample median is the arithmetic average of the two center values of a numerically ordered sequence of the  $K$  data points.

For notational simplicity, we drop the explicit  $k$  dependence from the random variables  $z$ ,  $x$ ,  $y$ , and  $\omega$ . It is first shown that the statistical complex median of  $\omega$ , denoted by  $w_{\text{med}}$ , equals  $w_{\text{opt}}$ . If a priori knowledge of  $R_{xx}$  and  $r_{xz}$  is available, the optimal weight  $w_{\text{opt}} = R_{xx}^{-1}r_{xz}$  may be calculated. Define  $y_u = z - w_{\text{opt}}^*x$  to be a two-input canceller's output after optimal weighting is applied. Note that  $y_u$  is uncorrelated with  $x$  by definition. Solving for  $z$  yields,  $z = y_u + w_{\text{opt}}^*x$ , and substituting this into the quantity  $\omega = (z/x)^*$  results in

$$\omega = \left( \frac{z}{x} \right)^* = \frac{y_u^* + w_{\text{opt}} x^*}{x^*} = \left( \frac{y_u}{x} \right)^* + w_{\text{opt}} \quad (10)$$

Finding the statistical medians (denoted by med in small letters) of the real and imaginary parts of (10)

separately yields

$$w_{\text{med}_{r,i}} = \text{med}_{r,i} \left[ \left( \frac{y_u}{x} \right)^* + w_{\text{opt}} \right] = \text{med}_{r,i} \left[ \left( \frac{y_u}{x} \right)^* \right] + w_{\text{opt}_{r,i}} \quad (11)$$

where  $\text{med}_{r,i}$  refers to the statistical medians of the real ( $r$ ) and imaginary ( $i$ ) parts of the argument separately. Using (11) we may define  $w_{\text{med}} = w_{\text{med}_r} + jw_{\text{med}_i}$ . Similarly, the quantities  $w_{\text{opt}_{r,i}}$  refer to the real and imaginary components of  $w_{\text{opt}}$  separately, such that  $w_{\text{opt}} = w_{\text{opt}_r} + jw_{\text{opt}_i}$ . Since  $w_{\text{opt}_{r,i}}$  are constants, they may come out of the median function as in (11).

Next, it is shown that  $\text{med}_{r,i}[(y_u/x)^*] = 0$ , for the case where  $z$  and  $x$  are each zero mean, complex-Gaussian random variables (also true for any set of complex, symmetric densities associated with  $z$  and  $x$ ). The random quantity  $y_u$  is a linear combination of  $z$  and  $x$  and is therefore a zero mean, complex-Gaussian random variable. Since  $y_u$  and  $x$  are uncorrelated and Gaussian, they are independent. We normalize the quotient  $(y_u/x)^*$  (from (11)) by multiplying by the ratio of standard deviations as

$$b = \frac{\sigma_x}{\sigma_{y_u}} \left( \frac{y_u}{x} \right)^* = \left( \frac{\tilde{y}_u}{\tilde{x}} \right)^* \quad (12)$$

The quantities  $\sigma_x$  and  $\sigma_{y_u}$  are the standard deviations of  $x$  and  $y_u$ , respectively, and therefore  $\tilde{x}$  and  $\tilde{y}_u$  are normalized versions of  $x$  and  $y_u$ , respectively, each with unit variance. The pdf and cdf of  $b_{r,i}$  (subscripts  $r$  and  $i$  refer to the real or imaginary parts of  $b$ , separately) are derived in Appendix A, and results are given here, respectively, as

$$f_{b_{r,i}}(b_{r,i}) = \frac{1}{2(1 + b_{r,i}^2)^{3/2}}, \quad -\infty \leq b_{r,i} \leq \infty \quad (13)$$

and

$$F_{b_{r,i}}(b_{r,i}) = \frac{1}{2} + \frac{b_{r,i}}{2(1 + b_{r,i}^2)^{1/2}}, \quad -\infty \leq b_{r,i} \leq \infty. \quad (14)$$

(It can be shown that (13) is equivalent to the pdf of a random variable  $t' = t/\sqrt{2}$ , where  $t$  is another random variable having a Student's  $t$  density function with two degrees of freedom.) Because of the symmetries of  $f_{b_{r,i}}$ , it follows that  $\text{med}_{r,i}(b) = E\{b_{r,i}\} = 0$ , where  $E\{b_{r,i}\}$  denotes the expected value of the real ( $r$ ) and imaginary ( $i$ ) parts of  $b$ , separately. Thus, it follows that

$$w_{\text{med}} = E\{\omega\} = w_{\text{opt}} \quad (15)$$

The convergence performance of the adaptive weight  $w_{\text{MED}}$  is derived next.

## B. Convergence of $l_{\text{med}}$ Building Block: No Outliers

For adaptive implementations of the two-input MCC (i.e., for finite  $K$ ), we are interested in the order

statistic pdfs of  $b_{r,i}$ . These are given by [10]

$$f_p(b_{r,i}) = \frac{K!}{(p-1)!(K-p)!} F_{b_{r,i}}^{p-1}(b_{r,i}) [1 - F_{b_{r,i}}(b_{r,i})]^{K-p} f_{b_{r,i}}(b_{r,i}). \quad (16)$$

In order to determine the median order statistics of  $b_{r,i}$ , (13) and (14) are substituted into (16), where  $p_m = (K+1)/2$  is the median value (for  $K$  odd). It is found that the means of both of the median order statistics are zero, and the variances of both of the median order statistics are:  $\text{var}(\text{MED}(b_{r,i})) = 1/(K-1)$ , for  $K$  an odd integer and  $K > 1$ . These results are derived in Appendix B, and it is noted that the variances are only a (simple) function of the number of sample points,  $K$ . Thus, using (12) the variances of the real ( $r$ ) and imaginary ( $i$ ) parts of  $w_{\text{MED}}$  (from (9)), denoted as  $w_{\text{MED}_{r,i}}$ , are each

$$\text{var}(w_{\text{MED}_{r,i}}) = \frac{\sigma_{y_u}^2}{\sigma_x^2} \left( \frac{1}{K-1} \right). \quad (17)$$

Since the variances approach zero as  $K \rightarrow \infty$ , it follows that  $w_{\text{MED}} \rightarrow w_{\text{med}} = w_{\text{opt}}$  in the mean square sense for increasing  $K$ . The identical variances (17) are used in the following derivation of the analytical convergence rate.

The MOE often used to assess an  $N$ -input canceller's performance is the normalized output residue power (NORP) (also known as normalized MSE). It is normalized to the minimum residue power (i.e., minimum mean square error (MMSE)) achievable, and it is approximately equal to the inverse of the SINR performance metric for nonconcurrent processing [5, 11, 19]. (It is understood that the NORP, SINR, MSE, and MMSE MOEs are ensemble-averaged quantities.) The relationship is shown concisely in [19] to be:  $\text{SINR} = 1/(\gamma\xi - 1)$ , where  $\xi$  is the MMSE of an MVDR adaptive processor and  $\gamma$  is the inverse of the desired signal variance. Thus, minimizing the NORP maximizes the SINR. This provides justification for directly comparing the SINR convergence MOE to the NORP convergence MOE.

For a two-input canceller, the NORP is defined as

$$\eta = \frac{E\{|z - w^*x|^2\}}{\text{res}_{\text{opt}}} \quad (18)$$

where  $w$  is a complex weight derived using a general performance criterion, and

$$\text{res}_{\text{opt}} = E\{|z - w_{\text{opt}}^*x|^2\} \quad (19)$$

is the optimal (minimum) residue power (i.e., MMSE) when using MMSE criterion for the weight yielding  $w_{\text{opt}} = R_{xx}^{-1}r_{xz}$ . If  $w = w_{\text{opt}}$ ,  $\eta$  achieves its minimum value of 1. However, for adaptive implementations,

$$w = w_{\text{opt}} + \tilde{w} \quad (20)$$

where  $\tilde{w}$  is some weight error relative to the complex constant  $w_{\text{opt}}$ . Using the definition  $y_u = z - w_{\text{opt}}^*x$  as well as (18) and (20),  $\eta$  and  $\text{res}_{\text{opt}}$  may be expressed, respectively, as

$$\eta = \frac{E\{|y_u - \tilde{w}^*x|^2\}}{\text{res}_{\text{opt}}} \quad (21)$$

and

$$\text{res}_{\text{opt}} = E\{|y_u|^2\} = \sigma_{y_u}^2. \quad (22)$$

The quantity  $\tilde{w}$  is independent of  $z$  and  $x$  due to the assumed use of nonconcurrent processing. Knowing this and the fact that  $y_u$  is uncorrelated with  $x$ , the numerator of (21) is found to be

$$E\{|y_u - \tilde{w}^*x|^2\} = \sigma_{y_u}^2 + E\{|\tilde{w}|^2\}\sigma_x^2. \quad (23)$$

Thus, substituting (22) and (23) into (21), results in

$$\eta = 1 + E\{|\tilde{w}|^2\} \frac{\sigma_x^2}{\sigma_{y_u}^2} = 1 + E\{\tilde{w}_r^2\} \frac{\sigma_x^2}{\sigma_{y_u}^2} + E\{\tilde{w}_i^2\} \frac{\sigma_x^2}{\sigma_{y_u}^2}. \quad (24)$$

Note that subscripts  $r$  and  $i$  refer to the real and imaginary components of  $\tilde{w}$ , respectively.

The random quantity  $w_{\text{MED}}$  in (9) was shown previously to converge to  $w_{\text{med}} = w_{\text{opt}}$ , and its real and imaginary components were each shown to have a variance given by (17). For the two-input MCC, by using (9), (11), (17) and the fact that the means (in addition to the medians) of  $w_{\text{MED}_{r,i}}$  equal  $w_{\text{opt}_{r,i}}$ , respectively, it is clear for  $l_{\text{med}}$  criterion that  $\tilde{w}_{r,i} = w_{\text{MED}_{r,i}} - w_{\text{opt}_{r,i}}$ , and so  $E\{\tilde{w}_r^2\} = E\{\tilde{w}_i^2\} = \text{var}(w_{\text{MED}_{r,i}}) = \sigma_{y_u}^2/[\sigma_x^2(K-1)]$ . Thus, from (24), the NORP convergence MOE for the  $l_{\text{med}}$  building block is found to be

$$\eta = 1 + \frac{2}{K-1} \quad (25)$$

for zero mean, complex-Gaussian inputs ( $z$  and  $x$ ). In comparison, the standard  $l_2$  block using the SMI algorithm converges just slightly faster on average for the same assumptions:  $\eta = 1 + [1/(K-1)]$ , as shown in [5]. But, as will be shown in the next subsection, the  $l_2$  block is not nearly as robust as the  $l_{\text{med}}$  block. Lastly, it is noted that the two-input  $l_{\text{med}}$  algorithm, like the two-input  $l_2$  (SMI) algorithm, has a convergence MOE shown here to be only a function of the number of samples,  $K$ , and thus is independent of the two-input external covariance matrix.

### C. $N$ -Input Median Cascaded Canceller

Replacing the  $l_2$  blocks with  $l_{\text{med}}$  blocks throughout a GSCC results in the cascaded canceller configuration shown in Fig. 3 for the  $N = 4$  case, and may be similarly expanded for any value of  $N$ . We refer to this general  $N$ -input configuration as an MCC. It was shown that the two-input  $l_{\text{med}}$  adaptive weight

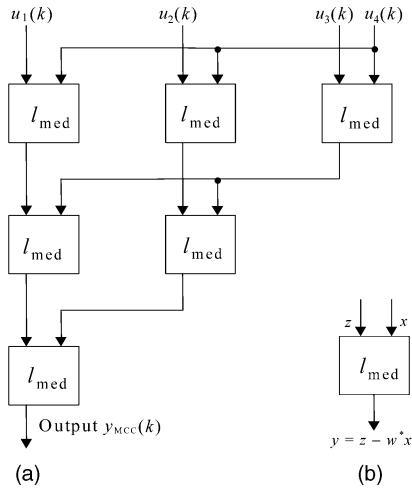


Fig. 3. (a) MCC with  $l_{med}$  building blocks for  $N = 4$  case. (b) Any single  $l_{med}$  building block canceller.

approaches the optimal two-input Gram-Schmidt weight as the number of samples  $K$  grows large. It is now argued, heuristically, that the set of adaptive cascaded weights for an  $N$ -input MCC approaches the optimal set of Gram-Schmidt cascaded weights as  $K$  increases. For each building block in an optimally weighted GSCC and for the corresponding block in an adaptively weighted MCC processor, let the difference between the optimal Gram-Schmidt weight and the adaptive MCC weight (9) be defined as a complex weight error  $e$ . Using Fig. 3 to illustrate the argument, we begin with the top row of building blocks. For the top row, each MCC building block was shown, using (15) and (17), to converge to the optimal Gram-Schmidt weight as  $K$  grows large, and with a weight error  $e$  that can be made arbitrarily small as  $K$  increases. Since the weight errors in the first row (or level) can be made arbitrarily small as  $K$  increases, the second level weight errors can be made arbitrarily small by originally choosing a larger  $K$  value. Hence, the 2nd level weights converge to optimal values (i.e., to the optimal Gram-Schmidt weights for the second level). This convergence argument can be repeated down to the  $(N - 1)$  or last level of weights. Thus, the  $N$ -input MCC adaptive weights converge to the  $N$ -input optimal Gram-Schmidt weights as  $K \rightarrow \infty$ .

Care must be taken when simulating or implementing an  $N$ -input MCC processor to handle cases when the sample medians from both terms in (9) happen to be from the same  $k$  value, and for when the MCC's original input data may have some real and/or imaginary components having zero value. For the case when both sample medians are from the same  $k$  value in (9), a weight is formed that produces a numerically zero value of output,  $y = z - w^*x$ , for the  $k$ th training sample. This zero is passed on to be either a  $z$  or  $x$  input to a follow-on two-input canceller within the  $N$ -input MCC structure. Due to this and/or any real/imaginary components having zero

value in the original input data, the quotients in (9) may produce undefined values due to forms such as  $a/0$  and  $0/0$ , where  $a$  is any real number. These undefined values are rare in practice. It has been found via simulation that if they do occur they may be either removed altogether before taking the sample median, or reassigned to be positive or negative extreme values, depending on the sign of  $a$ , whereupon they will have little effect on the sample median.

#### IV. CONVERGENCE RESULTS AND DISCUSSION

In this section simulation results are compared between the adaptive MCC and SMI processing algorithms in the presence of outliers. The NORP convergence performance is illustrated for a spatial adaptive array radar in a barrage jamming and noise environment. Following this, the SINR performance of the adaptive MCC and SMI STAP processors are compared for a simulated airborne radar in a representative nonhomogeneous land clutter, barrage jamming, and thermal noise scenario.

##### A. Spatial Adaptive Processing

In this subsection some representative convergence performance results are shown for an  $N$ -element, linear, spatial-only, adaptive array radar in a jamming and noise environment. The antenna elements have omni-directional element patterns and are spaced one-half of the transmitted wavelength apart. The receivers connected to each input channel (i.e., to each antenna element) are modeled as contributing to each channel spatially and temporally uncorrelated, complex-Gaussian, unit-variance noise. The spatial and temporal covariance matrices for receiver noise are identity matrices.

Barrage jammers are modeled as samples of narrowband, complex-Gaussian vector processes which are spatially correlated across the input channels of the array but are independent over discrete time  $k$ . A multiple jammer plus noise covariance matrix is formed using the methodology in [7]. Only jamming and noise (no desired signals (targets)) are modeled in the input interference-plus-noise covariance matrix. The Cholesky decomposition of the jamming and noise covariance matrix is used to create  $K$  random sample vectors or snapshots whose  $N$  elements have the desired covariance matrix. Snapshots are independent vector realizations of time-coincident jamming and noise complex voltages on the input channels. Because time corresponds to range in radar, time-coincident samples may be interpreted as range-coincident samples.

For this paper, lossless cascaded canceller structures are modeled; i.e., for the case of nonconcurrent processing and no desired signal

components in the training data, the canceller passes any input desired signal energy (if present) to the output with unity gain, while only correlated interference is removed at each stage or level. In general, a canceller's average NORP decreases monotonically to the minimum value achievable as more samples are used to estimate, and then cancel, the stationary interference. This behavior is illustrated in the results, but the effects of outliers are included as well. Specifically, convergence performance is compared between the SMI and MCC processors in the presence of a single and then multiple numbers of outliers added to the training data. It is shown that, depending on the magnitude of the outlier(s), their effects range from negligible to deleterious levels. Lastly, it is noted that all results shown in this subsection represent nonconcurrent processing methodology.

It was found that a target-like outlier in a snapshot of the training data is the most significant kind of outlier, and so most of the discussion is limited to this type. A target-like outlier is an outlier that is added only to the main channel ( $u_1$ ). This emulates the addition of a scaled,  $N \times 1$  desired signal or target steering vector for a canceller configuration,  $[\alpha \ 0 \dots 0]^T$ , where  $\alpha$  is a complex scalar and  $|\alpha|^2$  is the outlier or target power. We refer to this form as a target outlier vector. A general outlier vector refers to any configuration of a set of outlier(s) in a snapshot vector. In the following, it will be seen that even a single target outlier vector of sufficient power present in the training data slows significantly the convergence performance of the SMI processor but has practically no effect on the MCC processor. Simulations indicate this remains the case regardless of the effective rank of the interference-plus-noise covariance matrix. Effective rank refers to the number of eigenvalues of the interference-plus-noise covariance matrix that are large compared with the remaining smaller eigenvalues that are usually associated with receiver noise levels [23]. We use the shortened term "rank" to denote "effective rank" here.

In addition, it is found (results not shown) that adding a single, nontarget, complex-valued,  $N \times 1$  outlier vector consisting of either 1) random values, 2) a zero in the first or "main" element and generally non-zero random values in the remaining "auxiliary" elements, or 3) all zero values except one random value placed in an arbitrary auxiliary location, results in little or negligible degradation of the SMI convergence MOE for any outlier vector power level. Thus, a target-like outlier vector represents a worst case type of outlier vector for an SMI processor since it always produces a deleterious effect on convergence performance if it has sufficient power.

This little or negligible performance loss associated with a single, nontarget, outlier vector is generally observed in simulations unless the interference

environment is initially full rank to begin with, i.e., unless the number of narrowband jammers is approximately equal to  $N - 1$ . In this case a general outlier vector has a deleterious effect like a target outlier vector. With more than  $N - 1$  jammers, the cancellation performance will be very poor regardless of added outlier vectors. We say  $N - 1$  because of the narrowband rule of thumb for adaptive processors that one DOF is needed for the mainbeam constraint leaving  $N - 1$  DOFs for interference cancellation. For example, if  $N - 1$  spatially separated narrowband jammers are present and consequently nulled via the adaptive process, then no more DOFs are left to cancel the independent interference source represented by the high power outlier vector. Only for this full rank interference case does the addition of a single general outlier vector have an effect similar to a target-like outlier vector under any rank condition.

However, adding multiple random outlier vectors to an equivalent number of snapshots can saturate the available DOFs and cause generally poor jammer cancellation performance for the SMI processor. Multiple outlier vectors may simulate, for example, range-extended clutter that serves to corrupt the estimate of the jamming environment. Yet, the MCC processor proves to be relatively robust to a single or multiple numbers of outlier vectors of any power level, in the sense that significantly fewer training samples are needed for convergence compared with the SMI processor. This remains true even if the number of outlier vectors exceeds the number of DOFs (results not shown).

The Monte Carlo simulation is now outlined. Initially, a complex-Gaussian, narrowband, barrage jamming and noise covariance matrix is formed as described before. This covariance matrix is used to create  $K$  snapshot vectors of training data. For each Monte Carlo point  $K$ , a set of  $N_o$  independent, random, outlier vectors are created, each with unit norm (unit power). These vectors have one of the outlier vector configurations described above, depending on the goal of the simulation. They are subsequently scalar multiplied by the square root of the desired outlier vector power to adjust their powers to the desired level. Next they are added to the first  $N_o$  snapshots, up to a maximum of  $K$  snapshots (but typically much less than  $K$ ). Next, the group of  $K$  snapshots, now partially contaminated, is passed through the cascaded canceller and adaptively weighted at each stage.

An equivalent linear weight vector corresponding to the resultant set of cascaded canceller weights is calculated using the iterative procedure described in [18]. This equivalent weight vector is used to calculate the sample NORP (as described next). This NORP value is stored and eventually used in a 20 Monte Carlo average. This process is repeated for a range of outlier vector power levels spanning  $-10$  dB to



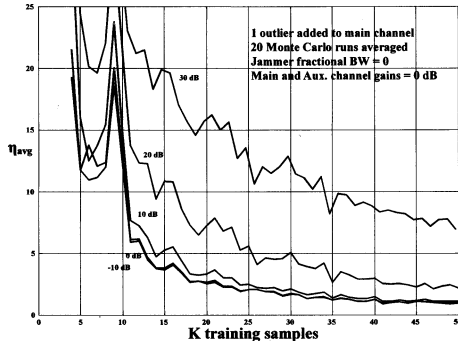


Fig. 4. Convergence plots of SMI processor with one target outlier vector added to main channel of training data having various power levels ( $-10$  to  $+30$  dB relative to noise).  $N = 10$  channels and one 20 dB sidelobe jammer modeled.

$+30$  dB relative to the common noise level in the receivers.

The processor's sample NORP in dB,  $\eta_{avg}$ , is plotted versus  $K$ , for several outlier vector power levels. Specifically, it is formed for each value of  $K$  as

$$\eta_{avg} = 10 \log_{10} \left[ \frac{1}{\text{res}_{opt}} \cdot \frac{1}{M_c} \sum_{i=1}^{M_c} \mathbf{w}_i^H \mathbf{R} \mathbf{w}_i \right] \quad (26)$$

where  $\mathbf{R}$  is the actual  $N \times N$  interference-plus-noise covariance matrix used to create the  $K$  input snapshot vectors,  $\mathbf{w}_i$  is the  $N \times 1$  equivalent linear adaptive weight vector (canceller weight vector) corresponding to the set of cascaded adaptive weights that were produced by the  $i$ th Monte Carlo input data matrix. The quantity  $\text{res}_{opt}$  is the expected optimal residue power output (MMSE) for the covariance matrix  $\mathbf{R}$  and the associated optimal MVDR weight vector. In (26) output power is normalized by  $\text{res}_{opt}$  so that as  $K$  increases the processor performance approaches 0 dB for any chosen interference and noise environment  $\mathbf{R}$ , facilitating comparisons. A value of  $M_c = 20$  is chosen since larger values result in little difference in the results. The MOE (26) expresses average convergence performance for nonconcurrent processing since no desired signal components are used in the formulation of  $\mathbf{R}$ .

Convergence curves for an SMI processor in a sidelobe canceller configuration (i.e., a GSCC) are shown in Fig. 4, for various powers of a single added target outlier vector. A total of  $N = 10$  channels are chosen, and one narrowband sidelobe barrage jammer and noise are modeled in the input data. The input jamming and noise data is considered to be in canceller form already. The jammer has a power equal to 20 dB above noise level. It is located at 26 deg clockwise from a direction normal to the array axis. Recall that the SMI algorithm convergence MOE [1] is approximately  $K = 2N = 20$  samples, which appears to be satisfied for plots corresponding to target outlier vector power levels  $-10$  dB to  $+10$  dB. However, as

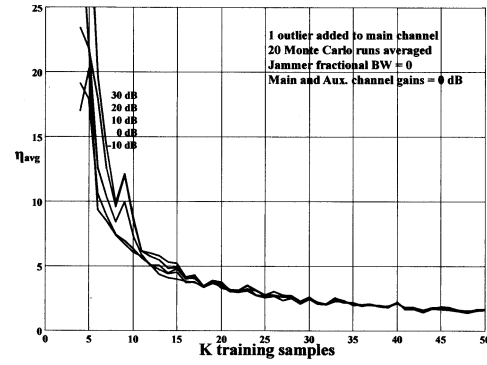


Fig. 5. Convergence plots of MCC with one target outlier vector added to main channel of training data having various power levels ( $-10$  to  $+30$  dB relative to noise).  $N = 10$  channels and one 20 dB sidelobe jammer modeled.

the target outlier vector power increases, it is evident that convergence slows significantly. For example, for a single  $+20$  dB target outlier vector, the convergence MOE is about 42 samples; for a single  $+30$  dB target outlier vector, many more than 50 samples would be required.

In Fig. 4 (as well as in Figs. 6 and 8), there exist peaks near  $K = N - 1 = 9$  samples. These are directly related to what we call a “bucket” effect that is due to the GSCC emulating the pseudo-inverse [20] of the input sample covariance matrix for  $K < N$ . It is known that the inverse of a sample covariance matrix is singular for cases when  $K < N$ . In such instances the pseudoinverse is often used instead to yield stable numerical results. In [8], it is shown this procedure produces an (inverse) bucket effect in plots of SINR versus  $K$  where the local minimum of a dip occurs at the same value  $K = N - 1$ .

For the MCC, also using  $N = 10$  and one 20 dB sidelobe jammer, Fig. 5 shows that convergence is essentially unaffected by the addition of a target outlier vector of any power level. Convergence is approximately equal to the ideal SMI convergence rate in pure Gaussian jammer and noise environments without outliers, showing near-optimal performance for this low value of input covariance matrix rank.

For five target outlier vectors present in the training data, Fig. 6 shows the SMI processor performs worse than the single outlier vector case shown in Fig. 4. For example, for five  $+20$  dB target outlier vectors (equal to the jammer level and therefore difficult to prescreen) the convergence MOE is again well over 50 samples. In comparison, it is seen in Fig. 7 that the MCC convergence MOE is degraded too, but only to about 30 samples (from about 22) for the worse case ( $+30$  dB) outlier curve. This represents that about 16% target outlier contamination (i.e.,  $5/30$ ) is accommodated, whereas the SMI processor cannot accommodate any target outlier contamination. In summary of Figs. 6 and 7, it is evident that strong desired signals (targets)

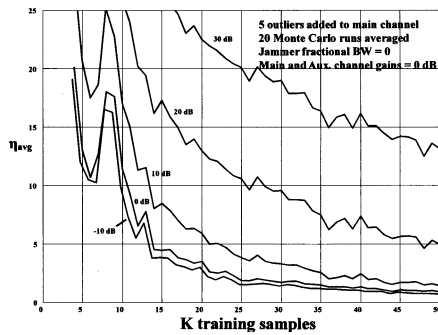


Fig. 6. Convergence plots of SMI processor with 5 target outlier vectors added to main channel of training data having various power levels ( $-10$  to  $+30$  dB relative to noise).  $N = 10$  channels and one 20 dB sidelobe jammer modeled.

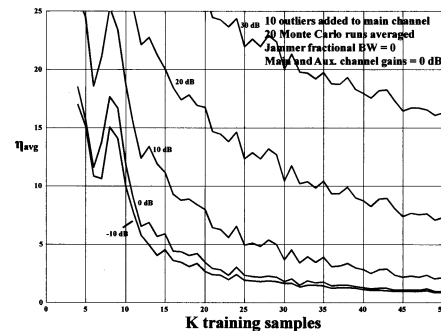


Fig. 8. Convergence plots of SMI processor with 10 target outlier vectors added to main channel of training data having various power levels ( $-10$  to  $+30$  dB relative to noise).  $N = 10$  channels and one 20 dB sidelobe jammer modeled.

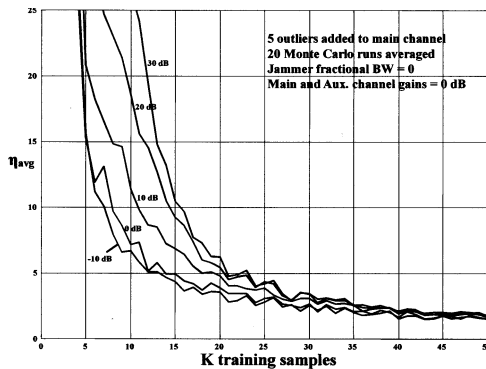


Fig. 7. Convergence plots of MCC with 5 target outlier vectors added to main channel of training data having various power levels ( $-10$  to  $+30$  dB relative to noise).  $N = 10$  channels and one 20 dB sidelobe jammer modeled.

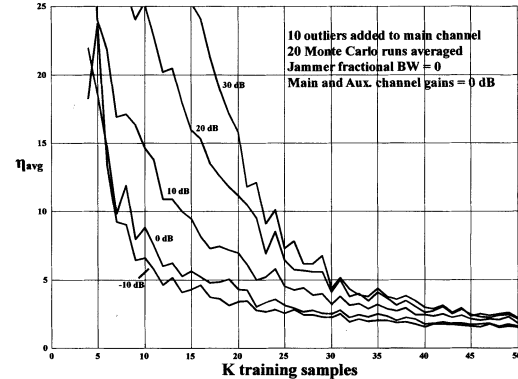


Fig. 9. Convergence plots of MCC with 10 target outlier vectors added to main channel of training data having various power levels ( $-10$  to  $+30$  dB relative to noise).  $N = 10$  channels and one 20 dB sidelobe jammer modeled.

in the training data result in little degradation of convergence performance for the MCC processor, but they have a deleterious effect on the SMI processor convergence. The MCC equivalent linear weight vector quickly approaches the optimum linear weight vector because the median functions in the  $l_{\text{med}}$  cancellers are relatively insensitive to the added desired signal vectors during weight estimation.

Figs. 8 and 9 show the effect of adding 10 target outlier vectors to the training data. The SMI processor convergence MOE is again poor. The convergence MOE for the MCC processor is approximately 40 samples (corresponding to 25% data contamination). Figs. 7 and 9 suggest that a significant amount (up to approximately 25%) of the training data may contain desired signals (targets) and/or target outlier vectors while achieving a 3 dB convergence MOE with only a modest increase in the number of training samples.

For Gaussian interference and noise with no general outlier vectors, it is well known that the SMI convergence rate is independent of the input interference-plus-noise covariance matrix for both the two-input and general  $N$ -input (Fig. 2) cases [1, 5]. Yet, for the MCC (Fig. 3), for the same assumptions, it appears from the simulation results that this strict

invariance to the input covariance matrix is limited to just the two-input case. Though still robust to outliers, as the number of interference sources (jammers) and their associated powers increase (i.e., as the effective rank of the interference-plus-noise covariance matrix increases) the convergence rate of the MCC degrades slowly and gracefully. It eventually becomes similar to an SMI processor performance with strong target outlier vectors present in the training data.

Examples of MCC convergence degradation as a function of increasing interference rank are shown in Figs. 10–13, using a different simulation scenario. Here we model  $N = 25$  half-wavelength spaced linear array input channels and up to four 30 dB barrage sidelobe jammers at fixed azimuth angles which are indicated in the plot captions. The received jammer powers are relative to receiver noise power. The following results indicate there is gradual performance degradation as the effective rank of the input interference-plus-noise covariance matrix (i.e., the number of narrowband jammers) increases.

Figs. 10–12 show the convergence performance of an  $N$ -input MCC under low rank conditions. It is similar to an SMI processor in pure Gaussian interference and noise for cases where the number

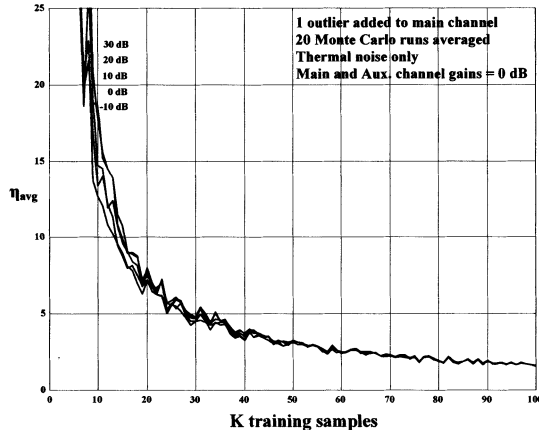


Fig. 10. Convergence plots of MCC with one target outlier vector added to main channel of training data having various power levels ( $-10$  to  $+30$  dB relative to noise).  $N = 25$  channels and no jammers (noise only) modeled.

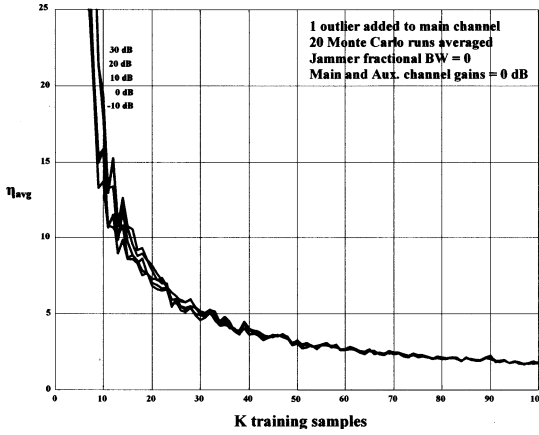


Fig. 11. Convergence plots of MCC with one target outlier vector added to main channel of training data having various power levels ( $-10$  to  $+30$  dB relative to noise).  $N = 25$  channels and one 30 dB sidelobe jammer at a 19.5 degree azimuth angle modeled.

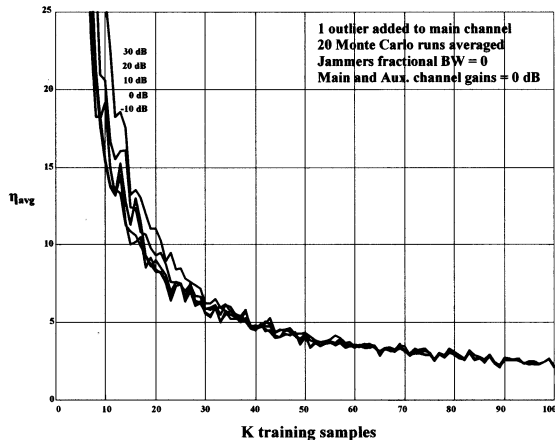


Fig. 12. Convergence plots of MCC with one target outlier vector added to main channel of training data having various power levels ( $-10$  to  $+30$  dB relative to noise).  $N = 25$  channels and two 30 dB sidelobe jammers at  $-8$  and 19.5 degree azimuth angles modeled.

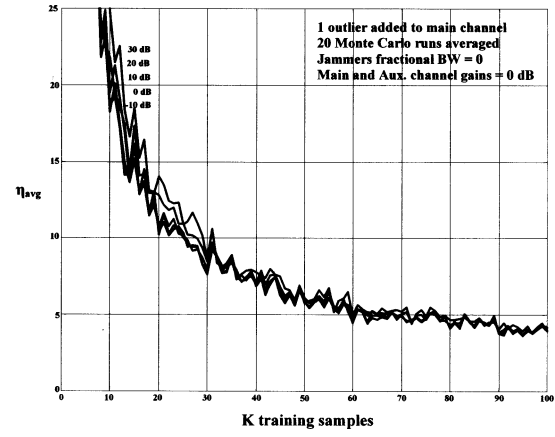


Fig. 13. Convergence plots of MCC with one target outlier vector added to main channel of training data having various power levels ( $-10$  to  $+30$  dB relative to noise).  $N = 25$  channels and four 30 dB sidelobe jammers at  $-28.2$ ,  $-8$ , 19.5 and 33.7 degree azimuth angles modeled.

of discrete narrowband interference sources is small (zero to two for this scenario) compared with the total number of DOF ( $N = 25$ ). Two out of 25 represents an 8% effective rank, and for this simulation it appears to be the rank threshold before there is a perceptible indication that convergence is becoming dependent on the interference-plus-noise covariance matrix. This is illustrated by increasing the number of jammers to four as shown in Fig. 13. Though still robust to outlier vectors, MCC convergence performance now begins to degrade slightly (i.e., curves rise) for increasing rank. This upward trend continues when adding more jammers (results not shown).

In summary, for pure Gaussian, stationary interference and noise environments, the convergence performance of the MCC is commensurate with SMI methods (and seemingly independent of the interference-plus-noise covariance matrix) for relatively low rank interference (roughly 8% of full rank). Performance degrades gracefully with increasing interference rank. However, one can infer from the plots shown so far that the effects of even just few outliers in the training data can degrade performance of the SMI processor to a level much worse than an MCC processor with the same input data. This superior performance may occur even though the MCC may be experiencing some loss in performance due to an increased interference rank, because it is highly robust to outliers.

## B. Space-Time Adaptive Processing

Convergence performance results are now presented for a linear array STAP processor using  $N$  elements in space and  $M$  pulses in time. The elements are spaced approximately one half wavelength apart and the pulses are spaced in time by the radar

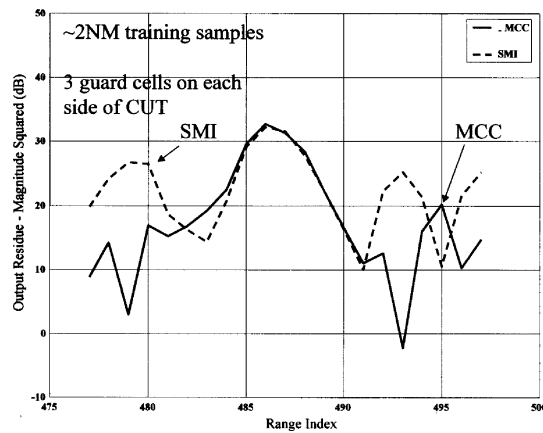


Fig. 14. Comparison of performance using SMI and MCC adaptive processors and RLSTAP-generated clutter. Guard cells and CUT excluded from weight training data. Target radar cross section = 10 dBsm. Two 30 dB sidelobe jammers modeled. Approximately 2NM samples used in weight training. SMI PTPN value is 5.5 dB. MCC PTPN value is 12.5 dB.

transmit waveform's pulse repetition interval (PRI). The simulation is configured as an airborne STAP radar processor in a clutter, jamming, and noise environment.

Due to platform motion, the return clutter echo from an airborne radar is spread in (Doppler) frequency. Since returns come from all azimuth angles due to antenna sidelobes and even backlobes, the clutter echo obscures (moving) targets having the same Doppler frequencies. The received echo has a two-dimensional or azimuth/Doppler spectral representation. For a particular range cell, the adaptive linear filter estimates weights that generally place nulls within this two dimensional spectrum in the directions of clutter and jamming energy. It does this while keeping a unity gain constraint in the target's azimuth/Doppler direction in an attempt to mitigate the effect of radar platform motion [7]. Optimal values of these weights maximize SINR at the filter output that then maximizes the probability of detection ( $P_d$ ) for pure Gaussian statistical environments [2]. Adequately estimating the optimum weights using experimentally measured data has been a challenge to the research community (especially in areas where nonhomogeneous land clutter is present), and no fully acceptable technique has been developed to date.

The MCC is used next as a STAP processor to filter out simulated ground clutter and jamming signals as received by a moving airborne monostatic radar. An example of this is shown in Figs. 14 and 15, where simulated airborne clutter data is used which is generated by a software model called RLSTAP (Research Laboratory Space Time Adaptive Processing), developed under contract via the Air Force Research Laboratory in Rome, NY [16]. Two 30 dB jammers (relative to noise level) are added to the data provided by RLSTAP. RLSTAP produces

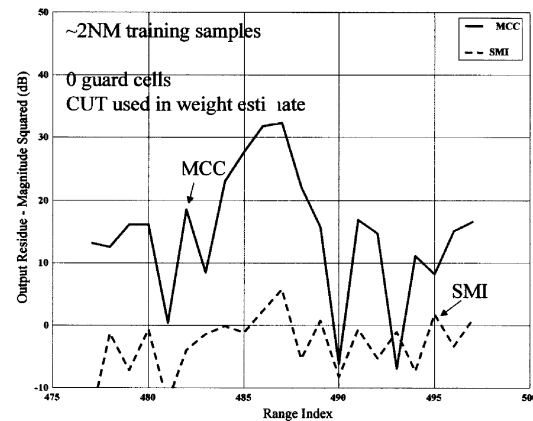


Fig. 15. Same scenario as in Fig. 14 except guard cells and CUT now used in weight training data. SMI PTPN value is 4 dB. MCC PTPN value is 14 dB.

validated site-specific clutter radar data for the areas where United States Geological Survey (USGS) Digital Terrain Elevation Data (DTED) and Land Use Land Cover (LULC) data is available. The terrain at White Sands, NM is chosen to provide representative nonhomogeneous land clutter scenarios, and STAP is expected to be required for good radar performance here. The terrain is a mixture of flat desert lands and mountainous regions in close proximity to one another. RLSTAP provides a site-specific radar reflectivity and terrain-shadowing map (not shown) and indicates that much radar shadowing exists in the mountainous regions due to the relatively low incident angle of the transmitted radar pulse typical of airborne surveillance radars. In addition, the power profile versus range of any of the received data channels appears spiky (10–20 dB swings) and generally irregular, even over typical short training data window widths of, for example, 220 range samples centered on the target. From this perspective we conclude the simulated terrain produces generally nonhomogeneous clutter.

The simulated radar is flying at 7000 meters altitude above ground level at 250 m/s going true north, while operating at a UHF frequency of 435 MHz and transmitting a peak power of 140 kW. The 14 antenna array elements are spaced just less than half wavelength apart (i.e., 0.3331 m), and the antenna boresight is pointed exactly north-west, so as to point directly towards a nonfluctuating 10 square meter (10 dBsm) radar cross section moving target. Thus, a 45 deg crab angle (due to a mechanically rotated antenna in this case) is simulated. The target velocity components are 100 m/s south and 100 m/s east (flying directly towards the radar along boresight), at an altitude of 2500 m and a range of 70.7 km north-west of the radar. The pulse repetition frequency (PRF) is 625 Hz, and 32 pulses per coherent processing interval (CPI) are collected (though only 8 are used in the analysis). There are

829 range cells of data collected in the CPI to provide sample support for the adaptive weight training.

It is desired to test how well the MCC, as a STAP processor, can pass target energy while rejecting clutter and jamming, both with and without using guard cells and the CUT in the training data. To do this we first transform the input data as shown in Fig. 1 to canceller form as follows. The  $NM \times K_t$  input data matrix  $\mathbf{Q} = [\mathbf{q}(1) \ \mathbf{q}(2) \dots \mathbf{q}(K_t)]$  is formed using  $K_t = K + 7$  snapshots that include  $K$  training data snapshots, six guard cell snapshots (three on each side of CUT), and the CUT snapshot. (For results that do not exclude guard cells and the CUT from the training data,  $K_t = K$ .) The data matrix  $\mathbf{Q}$  is then is premultiplied by a nonsingular  $NM \times NM$  unitary transformation matrix  $\mathbf{A} = \mathbf{T}^H$ . The matrix  $\mathbf{T}$  is found using the singular value decomposition (SVD) [22] of the  $NM \times 1$  normalized target steering vector  $\mathbf{s}$ , where  $\mathbf{s}$  has unit norm. Specifically, the SVD is formed as  $\mathbf{s} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^H$ , where the  $NM \times NM$  matrix  $\mathbf{U}$  and the scalar  $\mathbf{V}$  are unitary. The  $NM \times 1$  column vector  $\mathbf{\Sigma}$  has the lone singular value in the first element and remaining elements are zeros. The matrix  $\mathbf{T}$  can be formed as a matrix of column vectors where the first column vector is  $\mathbf{s}$  and the last  $NM - 1$  column vectors are the last  $NM - 1$  columns of  $\mathbf{U}$ . Thus,  $\mathbf{T} = [\mathbf{s} \ \mathbf{C}]$  where  $\mathbf{C} = \mathbf{U}(2, \dots, NM)$ . The  $NM \times (NM - 1)$  matrix  $\mathbf{C}$  is a basis for the left nullspace [22, p. 94] of  $\mathbf{s}$  as resulting from the SVD; i.e., it can be shown that  $\mathbf{C}^H \mathbf{s} = \mathbf{0}$ , where  $\mathbf{0}$  is an  $(NM - 1) \times 1$  zero vector.

Performance is compared in terms of the ratio of the peak target output power to the peak noise power—a measurement related to the instantaneous SINR—as measured within the data range window. We refer to this as the PTPN metric. In this context “peak noise power” refers to the peak ambient interference-plus-(thermal)-noise residue power output for ranges without targets. This metric (shown in dB scale) is chosen due to the expected constant false alarm rate (CFAR) type detection processing that normally follows canceller processing in real systems. In one form of CFAR processing, the average power in range cells neighboring the CUT, excluding a set of guard cells on either side of the CUT, is measured. This is used to set a detector threshold in order to declare target presence or absence. Thus, the PTPN metric is directly related to target detection probability.

Fig. 14 compares the outputs of the standard SMI STAP processor and MCC STAP processor for the case of using  $K = 220$  samples or approximately  $2 \times$  DOF of training data ( $\text{DOF} = N \times M = 112$  where  $N = 14$  channels and  $M = 8$  pulses). In this case, the CUT and guard cells are omitted from the training data as is standard practice with SMI-type processors to avoid canceling the target signal. Both the SMI and MCC processors indicate target presence with the PTPN value of the SMI processor equal to about

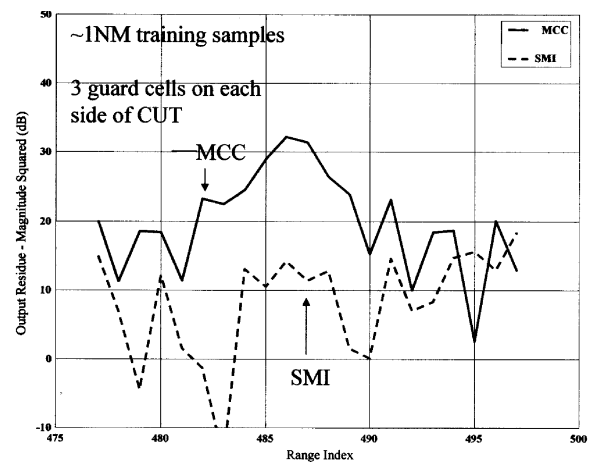


Fig. 16. Comparison of performance using SMI and MCC adaptive processors and RLSTAP-generated clutter. Guard cells and CUT excluded from weight training data. Target radar cross section = 10 dBsm. Two 30 dB sidelobe jammers crossed. Approximately  $NM$  samples used in weight training. SMI has no target peak. MCC PTPN value is 8.5 dB.

5.5 dB. The MCC performs better and has a 12.5 dB PTPN value. Peak noise levels are indicated using arrows in the figures.

Fig. 15 shows the case where the guard cells and the CUT are included in the training data used to form the adaptive weights—concurrent processing is used. In this case the SMI processor apparently experiences target signal cancellation of 26.5 dB, and results in a PTPN level of only 4 dB. The MCC experiences no target signal cancellation and actually improves in PTPN to 14 dB. It is conceivable that this improvement in the MCC performance is due to a better covariance matrix estimate of the CUT since the CUT and the (presumably) similar covariance structured guard cells are used directly in the weight estimate.

Since, the CUT and guard cells apparently can be included in the training data for the MCC processor with no loss of performance, the required computational requirements for real-time STAP applications are significantly reduced. The processor would only have to calculate one equivalent adaptive weight vector for an entire training region where it is felt clutter is homogeneous in nature. Then, that weight vector may be applied repeatedly to each range cell in the training region. For our example of 112 DOF with 220 training samples used, there could conceivably be a factor of 220 reduction in the real-time processing requirements. This feature may result in significant computational and cost savings for real systems.

In Figs. 16 and 17 we reduce the number of samples used for weight training to 120 or about  $NM$ . In Fig. 16, guard cells and the CUT are not included in the reduced amount of training data. The SMI processor has zero PTPN value performance as

expected for this low sample support. However, the MCC maintains some performance with a reduced PTPN value of 8.5 dB. Note that this performance characteristic (i.e., good convergence performance using less than  $2NM$  number of samples per input channel) is similar to that of reduced rank adaptive processors [8, 13]. In Fig. 17, guard cells and the CUT are included in the data used to form weight estimates—concurrent processing is used. The SMI processor output drops to the  $-80$  dB level and is noisy with zero PTPN value (again). However, the MCC processor still maintains good performance with a PTPN value of 10 dB. It appears from Fig. 17 that the MCC processor experiences a few dB of signal cancellation at this low sample support.

The robust performance of the MCC may suggest its use as an adjunct processor placed in parallel with an SMI processor. Outputs of each processor would be sent to separate detectors. If either or both detectors declare a target, then a target is formally declared. This would provide a system designer with good cancellation performance in benign environments due to the SMI processor, where convergence performance is fast and generally independent of the number and strength of external interference sources. The MCC processor may perform better in more stressing data environments with outliers, targets, and nonstationary data as long as the rank of the input interference-plus-noise covariance matrix is medium to low, relative to full rank. This low rank condition may often be expected, especially in scenarios where STAP processing is used. The parallel combination of processors/detectors may provide additional detection performance over that of just a single processor/detector for a larger set of data environments. However, if the measured data environment is generally nonbenign (contains mostly nonhomogeneous clutter, for example), then the robust MCC processor might be considered for use exclusively.

## V. CONCLUSIONS

An MCC is introduced for use as a general robust multichannel adaptive array signal processor. Robust processors are needed in many cases due to the presence of impulsive noise spikes (outliers), targets, and nonstationary data. Standard open loop adaptive processors often use the adaptive MVDR algorithm or its equivalent implementations (i.e., SMI methods) to estimate adaptive weights. However, the convergence performance of the SMI algorithm is known to be highly sensitive to even small numbers of outliers and targets in the training data, and it may be substantially degraded by the presence of nonstationary training data as well.

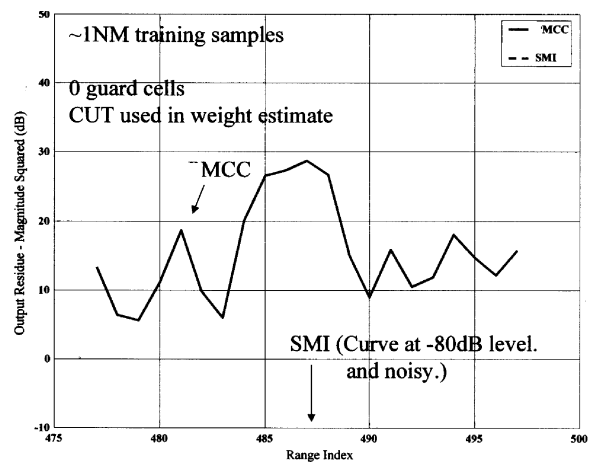


Fig. 17. Comparison of performance using SMI and MCC adaptive processors and RLSTAP-generated clutter. Guard cells and CUT now used in weight training data. Target radar cross section = 10 dBsm. Two 30 dB sidelobe jammers modeled. Approximately  $NM$  samples used in weight training. SMI has no target peak. MCC PTPN value is 10 dB.

Derived from the GSCC form, the MCC was shown to be a robust adaptive processor that uses the sample median function in each identical two-input building block within the MCC structure. For Gaussian interference and noise, the convergence performance for the basic building block canceller was derived and found to be independent of the two-input external interference and noise environment. It is only a function of the number of samples used for weight training like the SMI algorithm. Also, the two-input MCC convergence rate was shown to be commensurate with that of the two-input adaptive GSCC (SMI) algorithm but much more robust. The convergence independence property did not appear from simulations to extend to the general  $N$ -input case for pure Gaussian interference and noise. Convergence performance was shown to slowly degrade when increasing the effective rank of the interference-plus-noise covariance matrix (i.e., when increasing the number of narrowband jammers in a spatial adaptive array radar configuration).

However, for Gaussian data environments with a few added outliers (outlier vectors) and/or targets, convergence performance of an  $N$ -input SMI processor was shown to degrade significantly. For the  $N$ -input MCC processor, and using input data having a low rank input interference-plus-noise covariance matrix, nearly ideal convergence performance (i.e., near SMI performance with no outliers) is illustrated even in the presence of multiple strong outliers and/or targets in the training data. As the effective rank increased, the convergence performance was shown to degrade gracefully, but results showed that the MCC still outperformed SMI processors in outlier contaminated environments that may be typical of real-world data. Consequently, the MCC may be a

better choice of processor than an SMI type processor for many real world data environments.

Lastly, the MCC and SMI processors were compared in a STAP configuration for nonhomogeneous clutter, barrage jamming, and target-contaminated data. The clutter was generated using the hi-fidelity, site-specific, clutter simulation software called RLSTAP. The MCC outperformed the SMI processor in all cases in terms of PTPN value, an SINR-like metric associated with target detection performance when using standard range-cell averaging constant false-alarm rate (CFAR) techniques. This suggests that the MCC is more robust than an SMI processor to nonstationary data due to, in this case, nonhomogeneous clutter. This was shown to be true especially for cases where the guard cells and the CUT were included in the training data (where SMI processors experience significant cancellation of desired signal).

Due to this result, the amount of real-time calculations and associated hardware weight, volume, complexity, and cost that is required for an operational STAP radar may be reduced significantly by using an MCC processor. This feature of the MCC processor eliminates the need to compute a new adaptive weight vector for each range cell. Only one adaptive weight vector is required for an entire group of range cells where it is felt the clutter is homogeneous in nature. For the STAP example shown, this feature would reduce by 220 the required real-time computations in an operational STAP radar system. This target robustness feature also permits good detection performance in dense target environments. Lastly, the robust features of the MCC allowed good performance even when using as few as approximately  $1 \times \text{DOF}$  samples per input channel in the nonhomogeneous clutter STAP example. This is a convergence feature not shared by SMI type processors, and significantly aids the problem of limited sample support due to nonstationary data (e.g., due to nonhomogeneous clutter).

Results lead to a concept of operation where the MCC may be used in parallel with an SMI processor, with separate follow-on detectors. Both processors use the same input data, and if either or both follow-on detectors declare a target, a target is formally declared. This architecture would provide additional detection performance when the data environment becomes stressing due to multiple outliers and/or targets present in the training data, and if training data becomes substantially nonstationary. For certain applications that experience primarily nonbenign data environments containing outliers, targets, and nonstationary data, the MCC processor may always outperform SMI type processors. For this and cost reasons outlined earlier the MCC might be considered the sole choice of processor. More application-specific studies are needed to fully evaluate these issues.

## APPENDIX A. DERIVATION OF PDF AND CDF OF $b$

The derivation of the pdf and cdf of  $b$ , i.e., (13) and (14) respectively, can be shown by a series of transformations of random variables. Given that  $\tilde{y}_u$  and  $\tilde{x}$  have identical densities that are each zero mean, unit variance, complex-circular Gaussian (i.e., real and imaginary parts are joint IID and Gaussian) their (conjugate) quotient  $b$  can be put into polar form as

$$b = \left( \frac{\tilde{y}_u}{\tilde{x}} \right)^* = \frac{A_{\tilde{y}_u} e^{-j\phi_{\tilde{y}_u}}}{A_{\tilde{x}} e^{-j\phi_{\tilde{x}}}} = \frac{A_{\tilde{y}_u}}{A_{\tilde{x}}} e^{j(\phi_{\tilde{x}} - \phi_{\tilde{y}_u})} \quad (27)$$

where  $A_{\tilde{y}_u}$  and  $A_{\tilde{x}}$  are the amplitudes of  $\tilde{y}_u$  and  $\tilde{x}$ , and  $\phi_{\tilde{y}_u}$  and  $\phi_{\tilde{x}}$  are the phases of  $\tilde{y}_u$  and  $\tilde{x}$ , respectively. For any zero mean, circular complex-Gaussian random variable, its amplitude and phase are, respectively, Rayleigh distributed in the region  $[0, \infty)$  with parameter  $\alpha^2$  (which is related to the variance) and uniformly distributed in the region  $[-\pi, \pi)$ , and the amplitude and phase are also independent [14]. Thus, the amplitude and phase of  $\tilde{y}_u$  are independent, and the same is true for  $\tilde{x}$ . Since  $\tilde{y}_u$  and  $\tilde{x}$  are both uncorrelated by definition and complex-Gaussian, they are also independent. Thus, in (27),  $A_{\tilde{y}_u}$ ,  $A_{\tilde{x}}$ ,  $\phi_{\tilde{y}_u}$ , and  $\phi_{\tilde{x}}$  are independent of each other.

Equation (27) can be expressed in terms of two new random variables as,  $b = H e^{j\theta}$ , where  $H = A_{\tilde{y}_u}/A_{\tilde{x}}$  and  $\theta = \phi_{\tilde{x}} - \phi_{\tilde{y}_u}$ . Note that  $H$  and  $\theta$  are independent since their components are independent. The random variable  $H$  is the quotient of two Rayleigh random variables, and since  $A_{\tilde{y}_u}$  and  $A_{\tilde{x}}$  have equal Rayleigh parameter  $\alpha^2$ , it is shown next that  $H$  has density

$$f_H(H) = \frac{2H}{(H^2 + 1)^2}, \quad H \geq 0. \quad (28)$$

We derive (28) using the transformation of random variables technique, where auxiliary variables  $V$  and  $W$  are defined as  $V = g_1(A_{\tilde{y}_u}, A_{\tilde{x}}) = A_{\tilde{y}_u}/A_{\tilde{x}}$  and  $W = g_2(A_{\tilde{y}_u}, A_{\tilde{x}}) = A_{\tilde{x}}$ . Solving for  $A_{\tilde{y}_u}$  and  $A_{\tilde{x}}$  results in  $A_{\tilde{y}_u} = h_1(V, W) = VW$  and  $A_{\tilde{x}} = h_2(V, W) = W$ . Next, we substitute into the density formula for transformed variables [10],

$$\begin{aligned} f_{V,W}(V, W) &= f_{A_{\tilde{y}_u}, A_{\tilde{x}}}(h_1(V, W), h_2(V, W)) |J(V, W)| \\ &= f_{A_{\tilde{y}_u}, A_{\tilde{x}}}(VW, W) |W| \quad V, W \geq 0 \end{aligned} \quad (29)$$

then integrate out the auxiliary variable  $W$  to achieve the desired density. The symbol  $|\cdot|$  denotes magnitude and the Jacobian of the inverse transformation  $J(V, W)$  is

$$\begin{aligned} J(V, W) &= \det \begin{bmatrix} \partial A_{\tilde{y}_u} / \partial V & \partial A_{\tilde{y}_u} / \partial W \\ \partial A_{\tilde{x}} / \partial V & \partial A_{\tilde{x}} / \partial W \end{bmatrix} \\ &= \det \begin{bmatrix} W & V \\ 0 & 1 \end{bmatrix} = W \end{aligned} \quad (30)$$

where “det” denotes determinate. Since  $A_{\tilde{y}_u}$  and  $A_{\tilde{x}}$  are independent, their joint density is the product of

their marginal densities. The product of two identical Rayleigh densities is substituted into (29) as

$$f_{V,W}(V,W) = \left( \frac{1}{\alpha^4} V W^2 |W| \exp^{-W^2(V^2+1)/2\alpha^2} \right),$$

$$V, W \geq 0 \quad (31)$$

where,  $\alpha_{A_{\tilde{y}_u}}^2 = \alpha_{A_{\tilde{x}}}^2 = \alpha^2$ . We integrate out  $W$  in the region  $[0, \infty)$  to achieve the desired marginal density

$$f_V(V) = \int_0^\infty f_{V,W}(V,W) dW = \frac{2V}{(V^2+1)^2}, \quad V \geq 0 \quad (32)$$

and we note this is the form in (28) for the density of  $H$ . Using the substitution of variables  $g = V^2 + 1$ , one can easily show (32) integrates to one for  $V \geq 0$ , and is therefore a valid pdf.

Finding the distribution of the real part of  $b$ , i.e.,  $H \cos(\theta)$ , requires one more transformation of random variables using the same auxiliary variable technique. Set  $v = g_1(H, \theta) = H \cos(\theta)$ , noting  $H \geq 0$ , and set  $w = g_2(H, \theta) = \theta$  then solve for  $H$  and  $\theta$ :  $H = h_1(v, w) = |v/\cos(w)|$  and  $\theta = h_2(v, w) = w$ . In the following, by allowing  $v$  and  $w$  to take on any combination of values, the density function of  $H$ , when written in terms of  $v$  and  $w$ , must be reduced by exactly one-half from that of (28). This is due to  $v$  and  $\cos(w)$  each having symmetric pdfs about zero, which has the effect of doubling the original probability of each value of  $H$ . The density function for the transformed variables is

$$f_{v,w}(v, w) = f_{H,\theta}(h_1(v, w), h_2(v, w)) |J(v, w)|$$

$$= f_{H,\theta}(|v/\cos(w)|, w) |1/\cos(w)|,$$

$$-\infty < v < \infty, \quad -\pi < w \leq \pi \quad (33)$$

where

$$|J(v, w)| = \left| \frac{1}{\cos(w)} \right|. \quad (34)$$

It is found that the random variable  $\theta = \phi_{\tilde{x}} - \phi_{\tilde{y}_u}$  is uniformly distributed in the region  $[-\pi, \pi)$  because  $\phi_{\tilde{y}_u}$  and  $\phi_{\tilde{x}}$  are independent and uniformly distributed in the region  $[-\pi, \pi)$ . Since  $H$  and  $\theta$  are independent, their joint density is the product of their marginal densities. Substituting the product of marginal densities into (33) results in

$$f_{v,w}(v, w) = \frac{|v|}{2\pi \cos^2(w) [(v^2/\cos^2(w)) + 1]^2},$$

$$-\infty < v < \infty, \quad -\pi < w \leq \pi. \quad (35)$$

The factor of 1/2 is included in (35) as mentioned above to correct for the probability doubling effect due to the allowance of any combination of  $v$  and  $w$  used to define the density of  $H$ . Equation (35) may be

reduced to

$$f_{v,w}(v, w) = \frac{|v|}{2\pi} \cdot \left( \frac{\cos^2(w)}{\cos^4(w) + 2v^2 \cos^2(w) + v^4} \right),$$

$$-\infty < v < \infty, \quad -\pi < w \leq \pi. \quad (36)$$

Since the density function (36) is even with respect to  $w$ , i.e.,  $f_{v,w}(v, w) = f_{v,w}(v, -w)$ , when integrating out  $w$  we may reduce the integration space by half ( $0 \leq w \leq \pi$ ), and make up for it by multiplying by two as

$$f_v(v) = \frac{|v|}{\pi} \int_0^\pi \frac{\cos^2(w)}{\cos^4(w) + 2v^2 \cos^2(w) + v^4} dw,$$

$$-\infty \leq v \leq \infty. \quad (37)$$

Since  $\cos^2(\pi - w) = \cos^2(w)$ , it also follows that

$$f_v(v) = \frac{2|v|}{\pi} \int_0^{\pi/2} \frac{\cos^2(w)}{\cos^4(w) + 2v^2 \cos^2(w) + v^4} dw,$$

$$-\infty \leq v \leq \infty. \quad (38)$$

It can be shown that the solution to the integral portion of (38) is

$$\int_0^{\pi/2} \frac{\cos^2(w)}{\cos^4(w) + 2v^2 \cos^2(w) + v^4} dw = \frac{\pi}{4|v|(1+v^2)^{3/2}}.$$

$$(39)$$

Substituting (39) into (38) results in the same form as (13). The mean and variance of  $v$  can be shown to be zero and infinite, respectively. The cdf of  $v$  is found to match (14) by integrating the pdf from  $-\infty$  to  $v$ . Through a similar procedure as (33)–(39), the imaginary part of  $b$ ,  $H \sin(\theta)$ , can be shown to have identical density and distribution functions as the real part of  $b$ .

## APPENDIX B. DERIVATION OF MEDIAN ORDER STATISTICS

The density and distribution functions (13) and (14) are substituted, respectively, into the order statistics density formula (16), expressed here with  $v$  substituted for  $b_r$  or  $b_i$ . Assuming  $K$  odd for simplicity, the median of  $K$  ordered samples is located at the  $p_m = (K+1)/2$  position. Substituting this value of  $p_m$  into (16) results, after some algebra, in the median order statistic density

$$f_{(K+1)/2}(v) = \left( \frac{K!}{2^K [(K-1)/2]!^2} \right) \left( \frac{1}{(1+v^2)^{(K+2)/2}} \right).$$

$$(40)$$

Note that for  $K = 1$ , the density (40) is equal to (13), as expected. Also, (40) integrates to one for odd values of  $K$ .



It can be shown that the first and second moments of the random variable  $v$  described by density (40) are, respectively,

$$E\{v\} = \int_{-\infty}^{\infty} v f_{(K+1)/2}(v) dv = 0 \quad (41)$$

and

$$\begin{aligned} E\{v^2\} &= \int_{-\infty}^{\infty} v^2 f_{(K+1)/2}(v) dv \\ &= \left( \frac{K!}{2^K [\{(K-1)/2\}!]^2} \right) \int_{-\infty}^{\infty} \frac{v^2}{(1+v^2)^{(K+2)/2}} dv \\ &= \left( \frac{K!}{2^K [\{(K-1)/2\}!]^2} \right) \left( \frac{\sqrt{\pi}}{2} \frac{\Gamma\left(\frac{K-1}{2}\right)}{\Gamma\left(\frac{K+2}{2}\right)} \right) \\ &= \frac{1}{K-1}, \quad K \text{ odd and } K > 1 \end{aligned} \quad (42)$$

where (42) is also the variance  $\sigma_v^2$  since the mean is zero, and  $\Gamma$  denotes the Gamma function. It is noted that  $\sigma_v^2 = \infty$  for  $K = 1$ . The last simplifying step in (42) involves Gamma function manipulations [15], where the initial substitution  $K = 2m + 1$  is made, and it is noted that the minimum of  $m = 1$  since the minimum of  $K = 3$  for defined values in (42), where  $m$  is a positive integer.

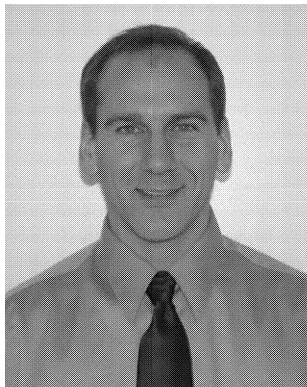
#### ACKNOWLEDGMENT

The authors thank Dr. Michael Steiner for providing a critical review of the material presented, and Dr. Eric Mokole for guidance in part of the mathematical development.

#### REFERENCES

- [1] Reed, I. S., Mallet, J. D., and Brennan, L. E. (1974) Rapid convergence rate in adaptive arrays. *IEEE Transactions on Aerospace and Electronic Systems*, **AES-10**, 6 (Nov. 1974), 853–863.
- [2] Brennan, L. E., and Reed, I. S. (1973) Theory of adaptive radar. *IEEE Transactions on Aerospace and Electronic Systems*, **AES-9**, 2 (Mar. 1973), 237–252.
- [3] Antonik, P., Schuman, H., Melvin, W., and Wicks, M. (1997) Implementation of knowledge-based control for space-time adaptive processing. In *Proceedings of IEEE Radar 97 Conference*, Oct. 14–16, 1997, 478–482.
- [4] Haykin, S. (1996) *Adaptive Filter Theory* (3rd ed.). Englewood Cliffs, NJ: Prentice-Hall, 1996.
- [5] Gerlach, K., and Kretschmer, F. F., Jr. (1990) Convergence properties of Gram-Schmidt and SMI adaptive algorithms. *IEEE Transactions on Aerospace and Electronic Systems*, **26**, 1 (Jan. 1990), 44–56.
- [6] Brennan, L. E., and Reed, I. S. (1980) Digital adaptive arrays with weights computed from and applied to the same sample set. In *Proceedings of 1980 Adaptive Antenna Symposium*, vol. 1, Dec. 1980, 236–249; Rome Air Development Center report RADC-TR-80-378.
- [7] Ward, J. (1994) Space-time adaptive processing for airborne radar. Technical report 1015, MIT Lincoln Laboratory, Lexington, MA, Dec. 13, 1994.
- [8] Steiner, M., and Gerlach, K. (2000) Fast converging adaptive processor for a structured covariance matrix. *IEEE Transactions on Aerospace and Electronic Systems*, **36**, 4 (Oct. 2000), 1115–1126.
- [9] Monzingo, R. A., and Miller, T. W. (1980) *Introduction to Adaptive Arrays*. New York: Wiley, 1980.
- [10] Papoulis, A. (1984) *Probability, Random Variables, and Stochastic Processes* (2nd ed.). New York: McGraw Hill, 1984.
- [11] Gerlach, K., and Kretschmer, F. F., Jr. (1991) Convergence properties of Gram-Schmidt and SMI adaptive algorithms, Part II. *IEEE Transactions on Aerospace and Electronic Systems*, **27**, 1 (Jan. 1991), 83–91.
- [12] Huber, P. J. (1981) *Robust Statistics*. New York: Wiley, 1981.
- [13] Goldstein, J. S., Reed, I. S., and Scharf, L. L. (1998) A multistage representation of the Wiener filter based on orthogonal projections. *IEEE Transactions on Information Theory*, **44**, 7 (Nov. 1998), 2943–2959.
- [14] McDonough, R. N., and Whalen, A. D. (1995) *Detection of Signals in Noise* (2nd ed.). New York: Academic Press, 1995.
- [15] Abramowitz, M., and Segun, I. (1968) *Handbook of Mathematical Functions* (5th ed.). Washington, D.C.: Government Printing Office, 1968, 255–256.
- [16] Maher, J., Callahan, M., and Lynch, D. (2000) Effects of clutter modeling in evaluating STAP processing for space-based radars. In *Proceedings of IEEE International Radar Conference*, 2000, 565–570.
- [17] Picciolo, M. L., Gerlach, K., and Goldstein, J. S. (2002) An adaptive multistage median cascaded canceller. In *Proceedings of IEEE Radar Conference*, Long Beach, CA, Apr. 22–25, 2002, 318–325.
- [18] Gerlach, K. (2001) Cascaded adaptive canceller using loaded SMI. *IEEE Transactions on Aerospace and Electronic Systems*, **37**, 2 (Apr. 2001), 710–719.
- [19] Goldstein, J. S., and Reed, I. S. (1997) Performance measures for optimal constrained beamformers. *IEEE Transactions on Antennas and Propagation*, **45**, 1 (Jan. 1997), 11–14.
- [20] Golub, G. H., and Van Loan, C. F. (1996) *Matrix Computations* (3rd ed.). Baltimore, MD: The Johns Hopkins University Press, 1996.

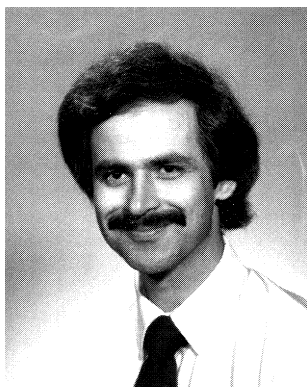
- [21] Picciolo, M. L., Gerlach, K., and Mili, L. (2002)  
Performance of a robust median cascaded canceller using measured MCARM airborne STAP radar data.  
In *Proceedings of Military Sensing Symposium: 48th Annual Tri-Service Radar Symposium*, Monterey, CA, June 24–28, 2002.
- [22] Strang, G. (1988)  
*Linear Algebra and its Applications* (3rd ed.).  
New York: Harcourt Brace Jovanovich, 1988.
- [23] Guerci, J. R., and Bergin, J. S. (2002)  
Principal components, covariance matrix tapers, and the subspace leakage problem.  
*IEEE Transactions on Aerospace and Electronic Systems*, **38**, 1 (Jan. 2002), 152–162.
- [24] Picciolo, M. L., and Gerlach, K. (2003)  
Median cascaded canceller using reiterative processing.  
In *Proceedings of the IEEE National Radar Conference*, Huntsville, AL, May 5–8, 2003, 71–78.



**Michael L. Picciolo** (S'87—M'88—SM'00) received a B.S. from Clarkson University, Potsdam, NY in 1988, an M.S. from Catholic University, Washington, D.C. in 1993, and a Ph.D. from Virginia Tech (Northern VA campus) in 2003, all in electrical engineering.

He currently works for SAIC in Chantilly, VA as an adaptive signal processing analyst. From 1988–2003 he worked for the Radar Division of the U.S. Naval Research Laboratory, Washington, D.C. He designed next generation surveillance radar systems and created novel adaptive signal processing algorithms. His interests include the field of robust statistics and the design of robust, fast-converging, adaptive algorithms. Application areas have included space-time adaptive processing (STAP) in airborne radar systems, advanced shipboard adaptive phased array radars, and radar sidelobe cancellers.

Dr. Picciolo serves as Chair of the IEEE Signal Processing chapter, Northern Virginia Section (1999–present), and also as the Section Treasurer for 2003.



**Karl Gerlach** (M'81—F'02) was born in Chicago, IL. He received his B.S. in 1972 from the University of Illinois, Urbana, and his M.S. and D.Sc. from George Washington University, Washington, D.C., in 1975 and 1981, respectively, all in electrical engineering.

Since 1972, he has been employed by the Naval Research Laboratory in Washington, D.C. From 1972 to 1976, he worked on experimental submarine communications systems and from 1976 to the present he has been with the Radar Division where his research interests include adaptive signal processing and space-based radar.

Dr. Gerlach was the 1986 recipient of the IEEE AESS Radar Systems Panel award.