Comparative study on PC-SD and MWF Algorithms for STAP RADAR

[1]Rizwana Fathima, [2]D. Venu
[1]M.Tech Student, Dept. of ECE,
Kakatiya Institute of Technology and Science, Warangal, Telangana, India,
[2]Assistant Professor, Dept. of ECE,
Kakatiya Institute of Technology and Science, Warangal, Telangana, India
[1]rizwanafathima256@gmail.com, [2]dunde.venu@gmail.com

***Abstract*—The paramount challenge in the radar system is to alleviate the consequences of cold (homogeneous) clutter, severe dynamic (heterogeneous) hot clutter and jamming interferences while estimating the states of targets under track. To surmount this challenge, Space-Time Adaptive Processing (STAP) intensify the competence of radar systems. Space-time Adaptive Processing is a two-dimensional filtering technique for antenna array with multiple spatially distributed channels. The name 'Space-Time' elucidate the coupling of multifarious spatial channels with pulse-Doppler waveforms. The term "Adaptive processor" signifies that it can employ using a variety of algorithms on many platforms ranging from space satellites to a small low flying unmanned aerial. In order to develop STAP algorithms to operate in adverse environments, where intense environmental interference can reduce STAP proficiency to detect and track ground targets. STAP can effectively suppress these interferences and maximize the signal to interference plus noise ratio (SINR). Methods such as principal component analysis, Multi-stage Weiner Filter (MWF) are applied to STAP system. Rank and Minimum square error are parameters considered for estimating the performance of two stated techniques.**

Keywords—STAP (space Time Adaptive Processing), homogeneous clutter, heterogeneous clutter, PC-SD, MWF, Rank, MSE.

I. INTRODUCTION

An adaptive processing uses spatial and temporal domains for signal detection ,which offers significant advantages in a variety of applications including radar, sonar, and satellite communication [1].The signal processing for radar systems uses multiple antenna elements that coherently process multiple pulses. An adaptive array of spatially distributed sensors, which processes multiple temporal snapshots, surmount the directivity and resolution limitations. Specifically, STAP using STAP creates an aptness to suppress interfering signals while simultaneously conserving gain of the desired signal. Additional gain afforded by an array of sensors leads to enhancement in the signal-to-noise ratio, resulting in an ability to place deep nulls in the direction of interfering signals.

Advanced airborne radar systems are equipped to detect targets in presence of both clutter and jamming. The ground clutter scrutinize by an airborne platform is extended in both angle and range and is spread in Doppler frequency because of the platform motion. STAP can significantly improve airborne radar performance. Computational complexity and the need to estimate non-stationary interference with limited data forces considerations of partially adaptive architectures. The STAP computational complexity is driven not just be the size of a single adaptive problem, but also by the number of adaptive problems that must be solved per coherent processing interval (CPI) [2].Fully adaptive STAP, though optimum given perfect knowledge, is impractical for two reasons. First is the computational burden of solving large system of equations in real-time. Second is the interference is unknown a – priori and must be estimated from the limited amount of data available during a radar dwell. The inherent non-stationary of radar clutter makes this estimation more difficult. Reduced dimension STAP Algorithms are required to ease both computation and training support. [3,4,5].

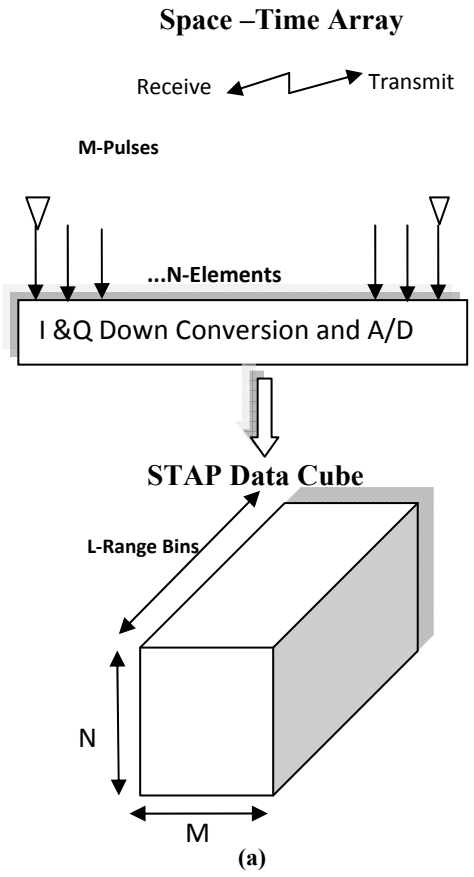
This paper utilizes the framework of space time adaptive processing for radar. In STAP, the sensor is composed of K elements and each element is followed by J taps spaced at the pulse repetition interval. The goal of this N=K J-dimensional STAP filter is to suppress clutter and interference for the purpose of improving target detection. In this paper, we propose that, how adaptive filtering of target signals can be achieved via Multi-stage wiener filter and principal component-signal dependent algorithms. Simulations will show that MWF generally offer pre-dominant rank and sample support compression than the more commonly used PC-SD. It is demonstrated that the new multistage wiener filtering technique provides a larger region of support for adaptivity.

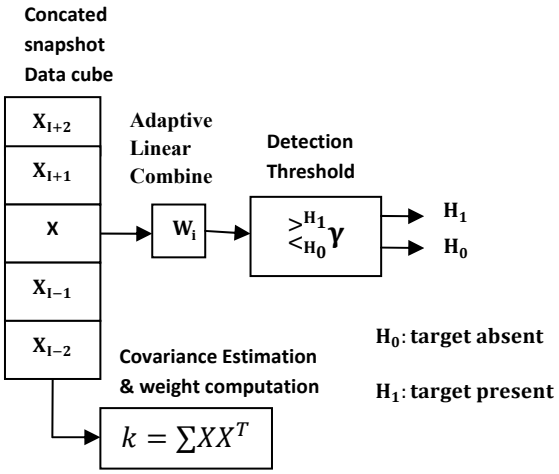
II. PROBLEM STATEMENT

Though SAR image tracking has much application appeal and performed admirably, it's not without shortcomings. Disadvantages to tracking with SAR image processing include delays from image synthesizing, overcoming typical image distortion effects and need for classifiers. For example, typical SAR image may need 3 to 5 seconds to be synthesized, therefore all target identification and tracking algorithms needs to wait at least that long. These factors create an environment where critical information is not produced in "real-time" which is crucial for fast decision making. This will explore an alternative – space-time adaptive processing (STAP).

It is capable of detecting target returns with information on target range and direction. It has many desirable traits: first, it is adaptive, meaning that it can function in any environment because it adapts to the environment. Second, STAP is faster than most alternatives such as SAR image tracking because it does not require heavy computations. Third, it is relativity simple meaning that it requires no specialized hardware. Fourth it is versatile to implement, meaning it can work using a variety of algorithms (such as PC and MWF) on many platforms ranging from space satellites to a small low flying unmanned aerial vehicle (UAV).We will provide the descriptions of how Reduced Rank Transformation like PC and MWF impact STAP in practice.

III. SIGNAL MODEL





(b)

Fig. 1. Description of STAP (a) STAP Data Cube. (b) Block diagram of STAP Algorithm.

Returns from a pulsed –Doppler radar are collected in a coherent processing interval(CPI) which can be represented by a 3-D data cube composed of N elements , J pulses and L range gates. The radar data is then processed at one K×J range gate of interest, which corresponds to a slice of the CPI data cube as shown in Fig (a). The data is then processed at one range of interest, which corresponds to a slice of the CPI data cube. This slice is a J ×K space-time snapshot whose individual elements correspond to the data from the jth pulse repetition interval (PRI) and the kth sensor element [2, 6, 7]. Hence this two-dimensional space-time data structure consists of element space information and PRI space-Doppler information. The snapshot is then stacked column-wise to form the KJ ×1 vector **x**. [6]

If a target is present in the range gate of interest, then the return is composed of components due to the target, the interference sources or jammers, clutter, and noise:

$$\mathbf{X} = \mathbf{X_i} + \mathbf{X_c} + \mathbf{X_n} \qquad (1)$$

If no target is present, then the snapshot consists only of interference, clutter, and white noise. The total input noise vector **n** is given by

$$\mathbf{n} = \mathbf{X}_i + \mathbf{X}_c + \mathbf{X}_n. \qquad (2)$$

Succinctly stated, most classical STAP algorithms consist of the following steps depicted in Figure (b).

(i) Estimate the parameters interference covariance matrix and target complex amplitude.

(ii) Form a weight vector based on the inverse covariance matrix

(iii) Calculate the inner product of the weight vector and the data vector from a cell under test

(iv) Compare the squared magnitude of the inner product in step (iii) with a threshold determined according to a specified false alarm probability. [1]

The input noise covariance matrix is then defined to be

$$\mathbf{R} = \mathbf{E}[\mathbf{n}\mathbf{n}^{\mathbf{H}}] \qquad (3)$$

Radar detection is a binary hypothesis problem, where hypothesis H_1 corresponds to target presence and hypothesis H_0 corresponds to target absence. Each of the components of the space-time snapshot vector **x** are assumed to be independent, complex, multivariate Gaussian. This snapshot, for each of the two hypothesis, is of the form,

$$\mathbf{H_0} : \mathbf{x} = \mathbf{n}$$
$$\mathbf{H_1} : \mathbf{x} = \mathbf{x_t} + \mathbf{n}. \qquad (4)$$

The KJ× 1-dimensional space-time steering vector v (ϑ1, ω1) is defined as follows:

$$\mathbf{v}(\boldsymbol{\vartheta}_\mathbf{t}, \boldsymbol{\omega}_\mathbf{t}) = \mathbf{b}(\boldsymbol{\omega}_\mathbf{t}) \otimes \mathbf{a}(\boldsymbol{\vartheta}_\mathbf{t}) \qquad (5)$$

Where b (ω_t) is the J×1 temporal steering vector at the target Doppler frequency ω_t and a (ϑ_t) is the K×1 spatial steering vector in the direction provided by the target spatial frequency ϑ_t. The notion (.) ⊗ (.) represents the Kronecker tensor product operator. For convenience in the analysis to follow, the normalized steering vector in the space-time look-direction is defined to be

$$s = \frac{V(\vartheta_t, \omega_t)}{\sqrt{V^H}(\vartheta_t, \omega_t)V(\vartheta_t, \omega_t)} \quad (6)$$

The two hypothesis in (4) may now be written in the form

$$\mathbf{H_0 : x = n}$$
$$\mathbf{H_1 : x = \alpha s + n}. \quad (7)$$

Where $\alpha = |\alpha| e^{j\varphi}$ is a complex gain whose random phase ϕ is uniformly distributed between 0 and 2π.The random vector **x**, when conditioned on ϕ , is Gaussian under both hypotheses. The conditional probability densities of **x** are

$$f_{X|H_{1,\varphi}}(\mathbf{x}) = \frac{1}{\pi^{KJ}\|\mathbf{R}\|} \mathbf{e}^{-(\mathbf{x}-\alpha\mathbf{s})^{H}\mathbf{R}^{-1}(\mathbf{x}-\alpha\mathbf{s})}$$
$$f_{X|H_{0,\varphi}}(\mathbf{x}) = \frac{1}{\pi^{KJ}\|\mathbf{R}\|} \mathbf{e}^{-\mathbf{x}^{H}\mathbf{R}^{-1}\mathbf{x}} \quad (8)$$

Where $\|(.)\|$ is the determinant operator. The likelihood ratio test then takes the form,

$$\Lambda = \frac{f_{X|H_1}(\mathbf{x})}{f_{\mathbf{x}|H_0}(\mathbf{x})} = \frac{\frac{1}{2\pi}\int_0^{2\pi} f_{X|H_1,\varphi}(\mathbf{x})\mathbf{d\varphi}}{\frac{1}{2\pi}\int_0^{2\pi} f_{\mathbf{x}|H_{0,\varphi}}(\mathbf{x})\mathbf{d\varphi}} \underset{H_0}{\overset{H_1}{\gtrless}} \eta \quad (9)$$

Where η is some threshold. Using the densities in (8), the test in (9) becomes

$$\Lambda = \mathbf{I_0}(2|\alpha||s^H R^{-1}\mathbf{x}|)e^{-|\alpha|^2 s^H R^{-1} s} \underset{H_0}{\overset{H_1}{\gtrless}} \eta \quad (10)$$

Where $\mathbf{I}_0(.)$ is the modified Bessel function of the first kind. The noise covariance matrix R is nonnegative definite and the modified Bessel function is monotonically increasing in its argument. Therefore, the test in (10) reduces to

$$\Lambda_1 = |s^H R^{-1}\mathbf{x}|^2 \underset{H_0}{\overset{H_1}{\gtrless}} \eta \quad (11)$$

Where the new threshold η_1 is related to the previous threshold η as follows :

$$\eta_1 = \left[\frac{\mathbf{I_0^{-1}}\left(\eta e^{|\alpha|^2 s^H R^{-1} s}\right)}{2|\alpha|}\right]^2. \quad (12)$$

The test in (11) was the first STAP detection criterion, developed in the well-known papers by Brennan and Reed [8] and Reed, Mallet and Brennan (RMB) [3].

1. PRINCIPAL COMPONENT-SIGNAL DEPENDENT :

Principal Component uses the Eigen-value decomposition (EVD) to produce a low rank estimate of the sampled covariance matrix $\widehat{\boldsymbol{R}}_{\mathrm{X}}$.[4, 9, 10, 11, 15, 16, 21]. This lower rank estimate would still be a good approximation to the original but it would dramatically reduce the required computer processing power. This " speed for accuracy" trade off is widely accepted in the industry due to benefit of reduced cost. Consider the MVDR-SMI beam former as follows:

$$\mathbf{W^{MVDR,SMI}} = \widehat{\mathbf{R}}_{\mathbf{X}}^{-1}\left(\frac{\mathbf{s}}{\mathbf{s^H}\widehat{\mathbf{R}}_{\mathbf{X}}^{-1}\mathbf{s}}\right) \quad (13)$$

$$\widehat{\boldsymbol{R}}_{\mathrm{X}} = \frac{1}{K}\sum_{k=1}^{K}\left[X_{i,n}(k)X_{i,n}^{H}(k)\right] \quad (14)$$

and K is the number of training snapshots. An EVD of $\hat{R}_X$ would be

$$\widehat{\boldsymbol{R}}_{\boldsymbol{X}} = \sum_{i=1}^{N} \lambda_i \mathbf{v}_i \mathbf{v}_i^{\mathbf{H}} \quad (15)$$

where $\boldsymbol{\lambda_i}$ and $\mathbf{v_i}$ represent the ith eigen value and eigen vector of $\widehat{\boldsymbol{R}}_{\mathrm{X}}$ and N is the total number of degree of freedoms.

The best reduced r rank approximation of $\hat{\boldsymbol{R}}_{\mathrm{X}}$ is formed by retaining the r largest eigenvalues and their corresponding eigenvectors and eliminate the rest. Therefore

$$\hat{\boldsymbol{R}}_{X}^{(PC)} = \sum_{i=1}^{r_{PC}} \lambda_i \mathbf{v}_i \mathbf{v}_i^H \qquad (16)$$

Where r_{PC} is less than N but contains the principal components of $\hat{R}_{\mathrm{x}}$ or the components with most signal power. Selecting the value for r_{PC} is to find the number of eigen values that are above the noise floor. One assumption from equation (16) is that the eigen value are ordered from highest to lowest. Meaning that the highest eigen value is at i =1, the next highest eigen value is at i=2, and so forth.

Principal component signal independent (PC-SI) algorithm is a data dependent signal independent rank reducing algorithm. The block diagram of PC-SI system is shown in figure: 2.

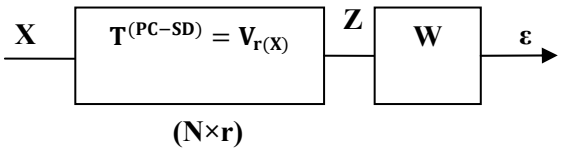


Figure: 2 Principal component signal independent

Principal component signal independent (PC-SI) algorithm is a data dependent signal independent rank reducing algorithm. It is considered to be data dependent because the data **x** is considered in weight vector calculation. It is considered signal independent because the steering vector **s** is not considered. One advantage of algorithm is that it is simple. A disadvantage is that we lose performance by not taking **s** into account. The PC-SI weight vector would be

$$\boldsymbol{W}^{\boldsymbol{MVDR,SMI,PC-SI}} = \left(\hat{\boldsymbol{R}}_{\mathrm{x}}^{\boldsymbol{PC}}\right)^{-1}\left(\frac{\boldsymbol{s}}{\boldsymbol{s}^{H}\left(\hat{\boldsymbol{R}}_{\mathrm{x}}^{\boldsymbol{PC}}\right)^{-1}\boldsymbol{s}}\right) \qquad (17)$$

Another principal component variation is Principal component signal dependent. The block diagram of PC-SD is shown in figure: 3.

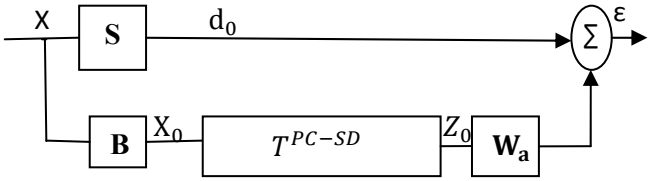


Figure 3: Principal component signal dependent

As the PC-SD take the steering vector s (or main lobe response) into account for rank reduction. The block B is a set of vectors that are orthogonal to s (side lobe responses).The steering vector for PC-SD would be

$$\mathbf{s}_{\boldsymbol{PC-SD}} = \mathbf{s} - \mathbf{B}\mathbf{w}_{\mathbf{a}} \qquad (18)$$

Where w_a is

$$\mathbf{W}_{\boldsymbol{a}} = \left(\hat{\boldsymbol{R}}_{\mathbf{X}}^{\boldsymbol{PC}}\right)^{-1}\mathbf{r}_{\mathbf{z}_0\mathbf{d}_0} \qquad (19)$$

and $\mathbf{r}_{\mathbf{z}_0\mathbf{d}_0}$ is teh cross correlation vector between $\mathbf{z_0}$ and $\mathbf{d_0}$ .The space –time weight vector would be

$$\boldsymbol{W}^{\boldsymbol{MVDR,SMI,PC-SD}} = \left(\hat{\boldsymbol{R}}_{\mathrm{x}}^{\boldsymbol{PC}}\right)^{-1}\left(\frac{\mathbf{s}_{\mathbf{PC-SD}}}{\mathbf{s}_{\mathbf{PC-SD}}^{H}\left(\hat{\mathbf{R}}_{\mathbf{X}}^{\boldsymbol{PC}}\right)^{-1}\mathbf{s}_{\mathbf{PC-SD}}}\right) (20)$$

2. MULTI-STAGE WEINER FILTER

Rank reduction for Weiner filter is heavily dependent on the cross correlation vector as

$$\mathbf{w}_{\mathrm{opt}} = \mathbf{R}_{\mathbf{x}_0}^{-1}\,\mathbf{r}_{\mathbf{x}_0 d_0} \qquad (21)$$

Where the weight vector is a function of both covariance matrix $R_{\mathbf{x}_0}$ and cross correlation vector $\mathbf{R}_{\mathbf{x}_0}$ and cross correlation vector $\mathbf{r}_{\mathbf{x}_{0\mathbf{d}_0}}$. Derived from the original weiner filter, the multi–stage weiner filter was introduced in [8].Its constrained form structure is shown in figure.4:

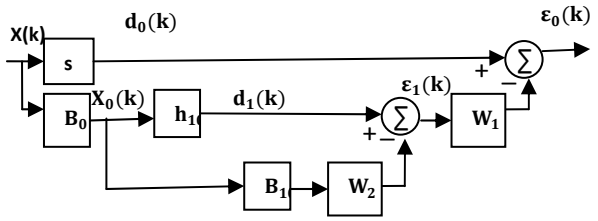


Fig.4: The filter structure of the MWF

In forward recursion ,the filter decomposes the sampled data snapshot **x** with a sequence of orthogonal projection like $\mathbf{B_0}$ [5].Rank reduction can be accomplished by truncating these decomposition stages to a desired number $\mathbf{r}^{\mathrm{MWF}}$.The result is a reduced rank transformation basis that spans the Krylov subspace instead of the eigenvector basis like PC.[12,17].

$$\mathbf{K}\left(\mathbf{s}, \mathbf{R}_{\mathbf{x}}, \mathbf{r}^{\mathbf{mwf}}\right) = \mathbf{span}\left\{\mathbf{s}, \mathbf{R}_{\mathbf{x}}\mathbf{s}, \mathbf{R}_{\mathbf{x}}^{\mathbf{2}}\mathbf{s}, \dots, \mathbf{R}_{\mathbf{x}}^{\left(\mathbf{r}^{\mathbf{mwf}}-\mathbf{1}\right)}\mathbf{s}\right\} \quad (\mathbf{22})$$

Since, it tailors its basis selection to the desired steering vector **s, the** MWF is able to operate in a more compact subspace than PC. [1]

After the forward recursion is completed, the MWF computes a series of scalar weights (w_1, w2, etc) at each stage and subsequently combine them to form the overall MWF weight vector.

$$\mathbf{W}^{\mathbf{mwf}} = \mathbf{s} - \mathbf{w_1}\mathbf{B}_{\mathbf{0}}^{\mathbf{H}}\mathbf{h_1} + \mathbf{w_1}\mathbf{w_2}\mathbf{B}_{\mathbf{0}}^{\mathbf{H}}\mathbf{B}_{\mathbf{1}}^{\mathbf{H}}\mathbf{h_2} - \mathbf{w_1}\mathbf{w_2}\mathbf{w_3}\mathbf{B}_{\mathbf{0}}^{\mathbf{H}}\mathbf{B}_{\mathbf{1}}^{\mathbf{H}}\mathbf{B}_{\mathbf{2}}^{\mathbf{H}}\mathbf{h_3} + \cdots \quad (\mathbf{23})$$

This technique has many desirable properties. First, its main computation operation is the simple vector cross correlation. Second, it does not form a covariance matrix which requires substantial computation work [18]. Last, it doesn't need matrix inversion or eigenvector decomposition, both of which are expensive operations [12, 19,20].

RANK COMPRESSION IN MWF:

MWF offer pre-dominant rank compression than PC. Received data covariance matrix can be decomposed into

$$\mathbf{R}_{\mathbf{x}_0} = \mathbf{V}\boldsymbol{\Lambda}\mathbf{V}^{\mathbf{H}} \sum_{\mathbf{i=1}}^{\mathbf{N}} \boldsymbol{\lambda}_{\mathbf{i}}\mathbf{v}_{\mathbf{i}}\mathbf{v}_{\mathbf{i}}^{\mathbf{H}} \quad (\mathbf{24})$$

MWF does not use the eigenvector basis, instead it uses the Krylov basis or

$$\boldsymbol{\varepsilon}\left(\mathbf{R}_{\mathbf{x}_0}, \mathbf{r}_{\mathbf{xd}}, \mathbf{r}^{\mathbf{mwf}}\right) = \mathbf{span}\left\{\mathbf{r}_{\mathbf{x_d}}, \mathbf{R}_{\mathbf{X_0}}\mathbf{r}_{\mathbf{x_d}}, \mathbf{R}_{\mathbf{x_0}}^{\mathbf{2}}\mathbf{r}_{\mathbf{x_d}}, \dots, \mathbf{R}_{\mathbf{x_0}}^{\mathbf{r}^{\mathbf{MWF}-\mathbf{1}}}\mathbf{r}_{\mathbf{x_d}}\right\} \quad (\mathbf{25})$$

Where $\mathbf{r}_{\mathbf{x}_\mathrm{d}}$ is the cross correlation vector between the data **x** and desired output d and $r^{\mathbf{mwf}}$ is the set rank [21].Expanding $\mathbf{r}_{\mathbf{x}_\mathrm{d}}$ we have

$$\mathbf{r}_{\mathbf{x_d}} = \boldsymbol{\alpha}_{\mathbf{1}}\mathbf{v_1} + \boldsymbol{\alpha}_{\mathbf{2}}\mathbf{v_2} + \cdots + \boldsymbol{\alpha}_{\mathbf{N}}\mathbf{v_N} \quad (\mathbf{26})$$

Where α_i is the cross correlation coefficient between the desired signal and the eigenvector $\mathbf{v_i}$. $\mathbf{r}_{\mathbf{x}_\mathrm{d}}$ is the first basis of the Krylov subspace. The second is $\mathbf{R}_{\mathrm{X}_0}\mathbf{r}_{\mathbf{x}_\mathrm{d}}$ .If we expand $\mathbf{R}_{\mathrm{X}_0}\mathbf{r}_{\mathbf{x}_\mathrm{d}}$ into

$$\mathbf{R}_{\mathbf{X_0}}\mathbf{r}_{\mathbf{x_d}} = \mathbf{V}\boldsymbol{\Lambda}\mathbf{V}^{\mathbf{H}}\left(\boldsymbol{\alpha}_{\mathbf{1}}\mathbf{v_1} + \boldsymbol{\alpha}_{\mathbf{2}}\mathbf{v_2} + \cdots + \boldsymbol{\alpha}_{\mathbf{N}}\mathbf{v_N}\right) \quad (\mathbf{27})$$

But, eigenvectors are orthonormal to each other except

with itself, or

$$\mathbf{v}_i^H\mathbf{v}_j = \{\frac{\mathbf{1\_if\_i = j}}{\mathbf{0\_if\_i \neq j}}$$

Thus, $$\mathbf{V}^{\mathbf{H}}\mathbf{r}_{\mathbf{x_d}} = [\boldsymbol{\alpha}_{\mathbf{1}}, \boldsymbol{\alpha}_{\mathbf{2}}, \dots \boldsymbol{\alpha}_{\mathbf{N}}]^{\mathbf{T}} \quad (\mathbf{28})$$

Equation (27) can be simplified to

$$\mathbf{R}_{\mathbf{X_0}}\mathbf{r}_{\mathbf{Xd}} = \boldsymbol{\alpha}_{\mathbf{1}}\boldsymbol{\lambda}_{\mathbf{1}}^{\mathbf{k}}\mathbf{v_1} + \boldsymbol{\alpha}_{\mathbf{2}}\boldsymbol{\lambda}_{\mathbf{2}}^{\mathbf{k}}\mathbf{v_2} + \cdots + \boldsymbol{\alpha}_{\mathbf{N}}\boldsymbol{\lambda}_{\mathbf{N}}^{\mathbf{k}}\mathbf{v_N} \quad (\mathbf{29})$$

Equations (26) and (29) are the first and second Krylov basis sets. The higher order Krylov basis can be defined as

$$\mathbf{R}_{\mathbf{x_0}}^{\mathbf{k}}\mathbf{r}_{\mathbf{xd}} = \boldsymbol{\alpha}_1\boldsymbol{\lambda}_1^{\mathbf{k}}\mathbf{v}_1 + \boldsymbol{\alpha}_2\boldsymbol{\lambda}_2^{\mathbf{k}}\mathbf{v}_2 + \cdots + \boldsymbol{\alpha}_N\boldsymbol{\lambda}_N^{\mathbf{k}}\mathbf{v}_N \qquad (30)$$

Where,

$$\mathbf{R}_{\mathbf{x_0}}^{k} = \mathbf{V}\boldsymbol{\Lambda}^{\mathbf{k}}\mathbf{V}^{\mathbf{H}} \qquad (31)$$

By observing at equation (30), we find that each of the Krylov vectors is a weighted sum of the eigenvectors. This is similar to principal components (PC).In fact if all α_i=1 the resulting rank compression is same as PC. Since these weight values are the function of both eigenvalue and the cross correlation coefficient then the MWF rank compression will always be better than or equal to its PC counterpart because $\alpha_i \leq 1$.

Therefore in Krylov subspace, if $\mathbf{r}^{\mathbf{MWF}} = N$ then all N Krylov basis vectors are kept and the full N-dimensional space is spanned. But if $\mathbf{r}^{\mathbf{MWF}} < N$ then the Krylov subspace dimension can be reduced based on low eigen value, low correlation, or a combination of both. In practice, it is observed that environment with low power interferers are well handled by MWF rank compression due to the low $\alpha_i\lambda_i^k$ product. Environments with closely spaced interference sources are also good candidate for MWF because their close proximity creates a bifurcation into a dominant eigenvector and a weak one. These weaker eigenvectors becomes additional candidates for rank compression by the MWF.

IV. STAP STIMULATION:

We examined PC and MWF rank compression for space-time adaptive processing (STAP). As mentioned earlier, STAP environment includes three types of undesirable interference signals: jammers, noise, and clutter. Figure.4: shows the Eigen spectra of two environments. One environment includes 2 randomly placed jammers of 30dB jammer to noise ratio (JNR), 10dB clutter to noise ratio (CNR), noise at 0dB and ICM effects. The second environment is the same as the first minus the jammers.

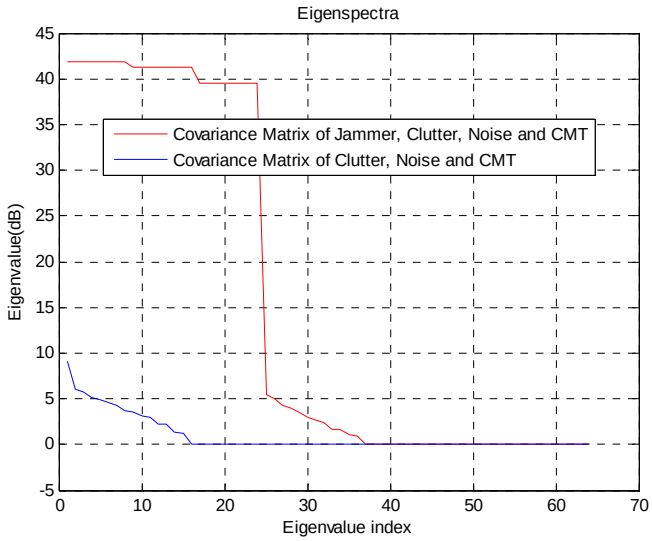


Figure.5: Eigen Spectra

This eigenspectra reveals the number of significant eigenvalues in the interference covariance matrix. The matrix without the jammers (blue curve) is dominated by clutter and this gives us perspective on the role clutter. The Brennan's rule or

$$\mathbf{r}_c = \mathbf{N} + (\mathbf{M} - \mathbf{1})\boldsymbol{\beta} \qquad (32)$$

It is a generally accepted guideline when dealing with clutter. It estimates the number of significant eigenvalues created by clutter with only three parameters: the number of elements N, the number of pulses M, and slope β. With our simulating parameters in Table.1.The calculated clutter rank is 17. This is near the simulated result of 18. The added rank could be the result of covariance matrix tapers (CMT). The second curve (red) demonstrates the impact of two 30dB JNR jammers on the eigenspectra. As jammer signals contaminates all channels. From simulation we see that contamination resulted in many more strong eigen values (i.e. rank). Comparing the two curves of Figure 5 we see that adding two jammers have doubled the rank and hence the number of needed adaptive degree of freedoms to cancel out the interference.

Therefore it is highly desirable to implement reduce rank transformations (RRT) to lower processing cost.

We evaluate the performance of MWF and PC-SD. simulation parameters are defined in Table 1 as follows.

Table.1 Simulation Parameters

PARAMETER	VALUE
N elements	8
M pulses /CPI	8
d -Inter element distance	λ/2
Clutter power(CNR)	10dB
Number of effective jammers	2
Jammer power (JNR)	50dB
Noise power	0dB
β (DPCA mode)	1
Monte Carlo trials	100
CMT type	ICM
B	5.7
wind speed	10Mph
PRF	1KHz
f_c (carrier frequency)	1 GHz

Figure .6: shows MWF and PC-SD performance against two jammers located at angles of [-72 23] degrees.

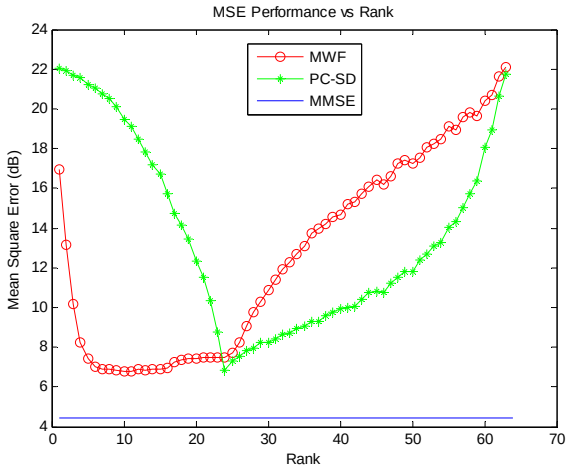


Figure 6: MWF Vs PC-SD 50dB JNR, 10dB CNR

PC-SD performance reaches lowest MSE at rank of 23; this means it needs 23 adaptive degrees of freedoms (ADoF) to suppress the interference to achieve MVDR (Minimum variance Distortion less Response). In contrast, MWF only needs 7 ADoF to accomplish the same. Notice that MWF also offers more flexibility in rank selection. As graph shows, MWF's MSE performance of ranks from 5 to 17 are all well within 3dB range of minimum mean square error (MMSE). This means that the MWF process can stop anywhere within stages 5 to 17 and still yield acceptable result. This type of flexibility is highly desirable.

By varying the JNR. While holding CNR at 10dB we decrease the JNR from 50dB (Fig 7 A.) to 20dB (Fig 7 D.). Figure 7 shows the MSE performances. Rank selection for PC-SD seems unaffected by the JNR changes, however MWF shows dramatic changes.

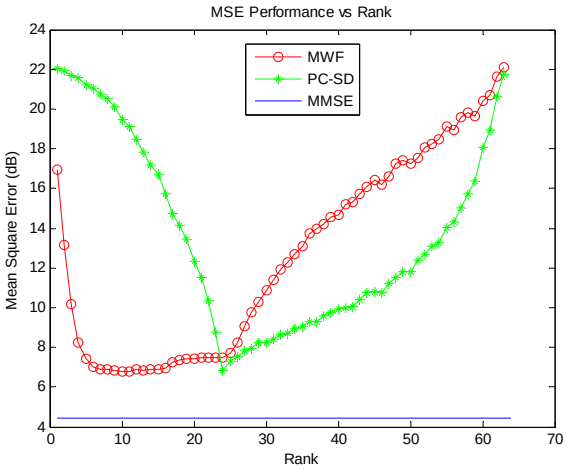
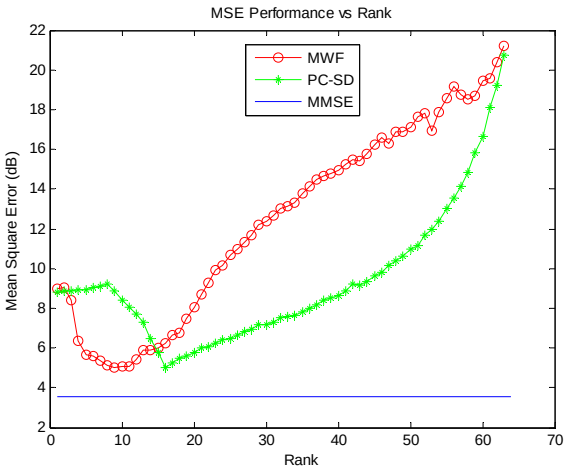
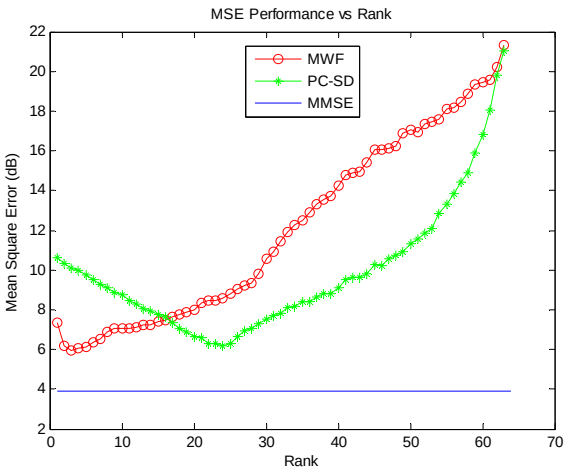

MSE Performance vs Rank
MWF
PC-SD
MMSE
Mean Square Error (dB)
Rank

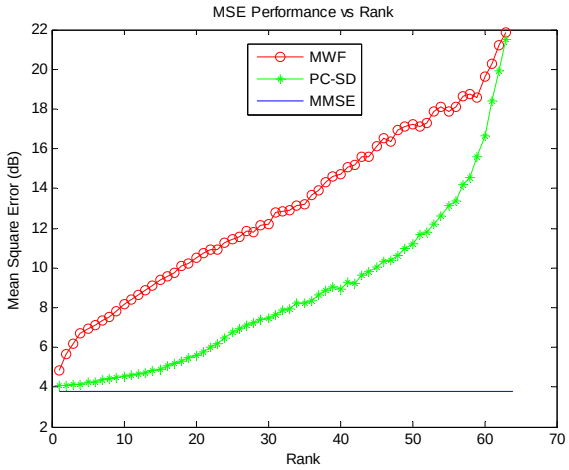
Fig.7(A)


MSE Performance vs Rank
MWF
PC-SD
MMSE
Mean Square Error (dB)
Rank

7(B)


MSE Performance vs Rank
MWF
PC-SD
MMSE
Mean Square Error (dB)
Rank

7(C)



7(D)

Figure.7: MWF Vs PC-SD for various JNR and Constant CNR. (A) JNR=50dB,CNR= 10dB. (B) JNR=40dB,CNR= 10dB. (C) JNR=30dB,CNR= 10dB. (D) JNR=20dB,CNR= 10dB

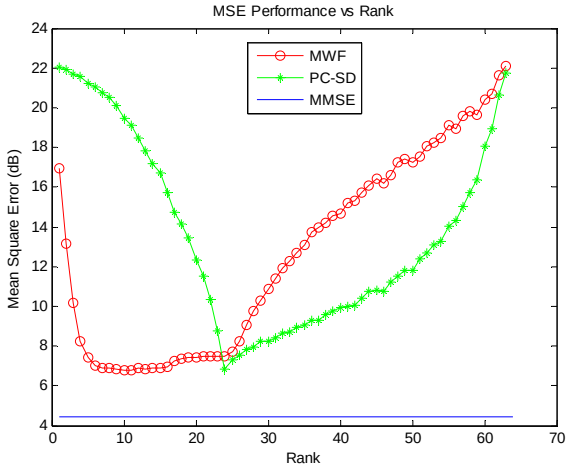
Table.2 shows the rank selections of MWF and PC-SD. MWF adapts to the interference levels and adjusts to its rank selection to received jammer power while PC-SD makes no adjustments. MWF adaptability in this case is desirable given that in practical situations the environment is constantly changing. In addition, MWF rank selections are less than its PC-SD counterparts which means that it could be done faster.

Table.2: RANK Selection for varying JNR

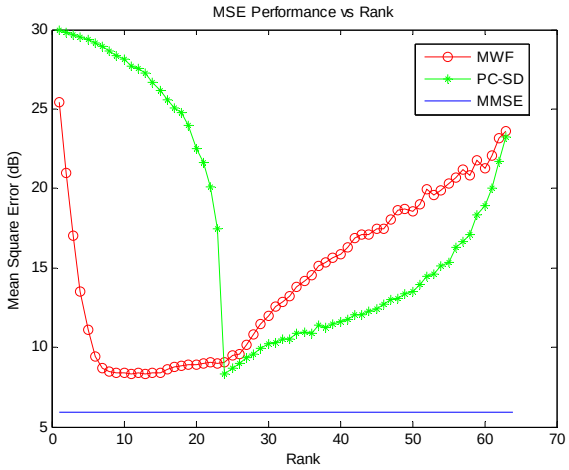
CASE	JNR (dB)	CNR (dB)	MWF Rank selection	PC-SD rank selection
A	50	10	5-17	16-30
B	40	10	3-16	16-30
C	30	10	1-12	16-24
D	20	10	1-5	9-17

Figure.8 and Table.3 shows rank selection for environments where CNR varies from 40dB (Fig 8 A.) to 10dB (Fig 8 D.) while JNR is constant at 50dB. In this case neither rank selection changes much, however MWF still offers lower rank selection.

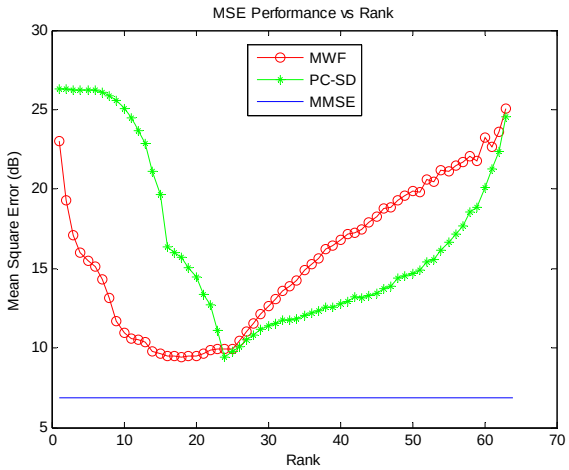
Figure.8: MWF Vs PC-SD for various CNR and JNR=50dB



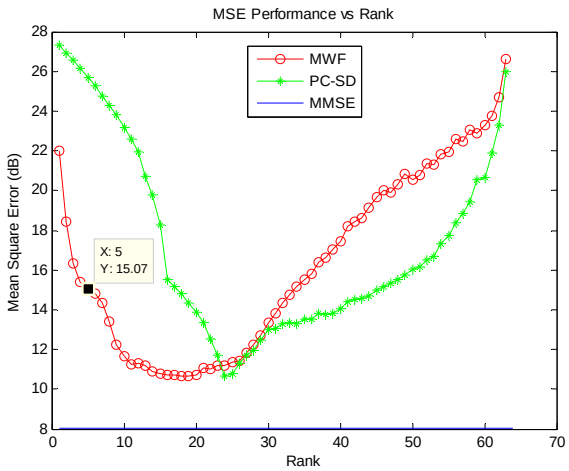
8(A)



8(B)



8(C)



8(D)

Figure.8: MWF Vs PC-SD for various CNR and Constant JNR. (A) JNR=50dB,CNR= 10dB. (B) JNR=50dB,CNR= 20dB. (C) JNR=50dB,CNR= 30dB. (D) JNR=50dB,CNR= 40dB.

Table 3 shows the rank selections of MWF and PC-SD. MWF adapts to the interference levels and adjust to its rank selection to received jammer power while PC-SD makes no adjustments. MWF adaptability in this case is desirable given that in practical situations the environment is constantly changing. In addition, MWF rank selections are less than its PC-SD counterparts which means that it could be done faster.

Table.3: RANK Selection for varying CNR

CASE	JNR (dB)	CNR (dB)	MWF Rank selection	PC-SD rank selection
A	50	10	5-17	16-30
B	50	20	5-17	16-24
C	50	30	4-20	16-24
D	50	40	5-19	16-21

CONCLUSION

In this paper we showed that MWF offered superior rank compression than PC-SD, especially in environments where jammer powers are lowered to 30dB. MWF demonstrated that it can adapt its rank selections to the environment but PC-SD did not. MWF did not significantly reduce sample requirements as hoped. In all instances, N×M samples were required to have an adequate estimate of the interference covariance matrix. There are two things we should clarify. First, generating the MSE performance graphs shown are not possible in practice. They are acquired in our simulation because we know exactly what the interference covariance matrix is in our simulated environment, but in practice that would require infinite number of samples which is practically impossible. As a result, optimum rank selection would be more or less "blind". Second, the majority of ranks did not achieve our desired minimum variance distortion less response (MVDR). In both PC-SD and MWF, MSE performance degraded further as ranks increased beyond the optimum rank. For case in Figure 6, only 12 out of 64 possible ranks yielded acceptable results. If we blindly select our process rank, the probability of failure would be 81%.

REFERENCES

[1] Muralidhar Rangaswamy, "An Overview of Space Time Adaptive Processing for RADAR", Proceedings of IEEE Radar Conference, 2003.
[2] James Ward, "Space-time Adaptive Processing for airborne Radar", IEEE Transactions on Aerospace and Electronic Systems, 1995.
[3] I.S. Reed, J.D. Mallet, and L.E. Brennan, "Rapid convergence rate in adaptive arrays", IEEE Trans. Aerosp. Electron. Syst., AES-10(6):853-863, November 1974.
[4] D.W. Tufts, R. Kumaresan, and I. Kristeins, "Data Adaptive Signal estimation by singular value decomposition of a data matrix. Proc., IEEE Trans. Aerosp. Electron. Syst., AES-70(6):684-685, June 1982.
[5] J. S Goldstein, I.S.Reed, and P.Zulch, "A Multistage Partially Adaptive STAP Detection Algorithm", IEEE Trans. Aerosp. Electron. Syst., 35(2): April 1999.
[6] J.Ward, "Space-Time adaptive Processing for Airborne Radar", Technical Report 1015, MIT Lincoln Laboratory, Lexington, MA, USA, December 1994.
[7] J.S. Goldstein, I.S.Reed, "Theory of partially adaptive Radar", IEEE Trans. Aerosp. Electron. Syst., 33(4): 1309-1325, 1997.
[8] L.E.Brennan and I.S.Reed. "Theory of Adaptive Radar". IEEE Trans. Aerosp. Electron. Syst., vol.9, pp: 237-251, March 1973.
[9] I.Kristeins and D.W. Tufts, "Adaptive Detection using a Low rank Approximation to a Data Matrix", IEEE Trans. Aerosp. Electron. Syst., 30(1):55-67, January 1994.
[10] E .K. L. Hung and R. M. Turner, "A Fast Beam forming Algorithm for Large Arrays", IEEE Trans. Aerosp. Electron. Syst., AES-19(4):598-607, July 1983.
[11] N.L. Owsley , Sonar Array Processing, In S. Haykin, editor, Array Signal Processing , Prentice- Hall, Englewood Cliffs, NJ , 1985.
[12] G. H. Golub and C.F. Van Loan, Matrix Computations , Johns Hopkins Univ.Press , Baltimore , MD, third edition , 1996.
[13] B.Friedlander , " A Signal Subspace Method For Adaptive Interference Cancellation", IEEE Trans. Acoust.,Speech , signal Processing, 36: 1835-1845 , December ,1988.
[14] A.M. Haimovich and Y. Bar-Ness.Adaptive antenna arrays using eigenvector methods", In Proc. IEEE ICASSP, pages 980-983, 1988.
[15] H. Cox and R.Pitre. Robust DMR and Multi-rate Adaptive Beam forming. In Proc.31st Asilomar Conf. Signals, Syst. Comput., vol.1, pp: 920-924, Pacific Grove, CA, November 1998.
[16] J.D. Hiemstra and J.S. Goldstein, "Robust Rank Selection for the Multi-stage Weiner Filter", In Proc. IEEE ICASSP, Orlando, FL, May 2002.
[17] M.L. Hoing and W.Xiao. Adaptive Reduced-rank interference Suppression with Adaptive rank selection. In Proc. MilComm 2000, vol.2, Los Angeles, CA, October 2000.
[18] D .C. Ricks and J.S. Goldstein, Efficient architectures for implementing adaptive algorithms. In Proc.2000 Antenna Applications Symposium, Allerton Park, Monticello, IL, September 2000.
[19] H.L. Van Trees. Optimum Array Processing .Wiley, Newyork, NY, 2002.
[20] J.S.Goldstein, I.S. Reed, and L.L.Scharf, "A Multistage representation of the wiener filter based on orthogonal projections", IEEE Trans. Information theory, 44(7):2943-2959, November 1998.
[21] J.S. Goldstein and I.S.Reed. Reduced rank adaptive filtering, IEEE Trans. Signal Processing, 45(2):492-496, February 1997.