

Multistage Adaptive Pulse Compression

Peter Vouras

Naval Research Laboratory
Washington, D.C. 20375

Abstract— Many modern radar systems employ pulse compression to maximize the energy on target while maintaining high range resolution. For a solitary point target in white noise, employing a matched filter on receive will maximize the target signal-to-noise ratio (SNR) at the output of the receiver. The matched filter itself is a time-reversed version of the transmitted waveform which is convolved with the received time series to pulse compress the data. A drawback to the matched filter receiver is the range sidelobes which extend on either side of the point target and may mask another weaker target. To reduce range sidelobes after pulse compression, novel adaptive pulse compression techniques have been developed. One such technique is the Reiterative Minimum Mean Square Error Adaptive Pulse Compression (RMMSE-APC) algorithm. This algorithm employs an optimal compression filter at each range bin and significantly reduces the range sidelobes in the vicinity of large targets. In this paper, a pulse compression filter with output identical to the RMMSE filter is derived by employing a multi-stage decomposition of the Wiener filter. A reduced rank version of the Multi-Stage Wiener Filter (MSWF) with lower computational complexity can be created by pruning the number of stages in the decomposition.

I. BACKGROUND

The signal received at the radar can be represented as a convolution of the transmitted signal and the impulse response of the range profile illuminated by the radar [1, 2]. Let the vector \mathbf{s} denote N discrete samples of the transmitted waveform $s(t)$, $\mathbf{s} = [s(0), s(1), \dots, s(N-1)]^T$, and let $\mathbf{x}(l) = [x(l), x(l-1), \dots, x(l-N+1)]^T$ represent a length N portion of the illuminated range profile for $l = 0, 1, \dots, L-1$, where l is a discrete delay index and L is the number of range cells of interest. Then the signal received at the radar is

$$\mathbf{y}(l) = \mathbf{A}(l)\mathbf{s} + \mathbf{v}(l) \quad (1)$$

where $\mathbf{v}(l)$ is an N -by-1 vector of additive noise samples. The noise is assumed to be temporally and spatially white and uncorrelated with the transmitted signal samples $s(k)$. The matrix \mathbf{A} is written as

$$\mathbf{A}(l) = \begin{bmatrix} \mathbf{x}^T(l) \\ \mathbf{x}^T(l+1) \\ \vdots \\ \mathbf{x}^T(l+N-1) \end{bmatrix} \quad (2)$$

$$= \begin{bmatrix} x(l) & x(l+1) & \cdots & x(l+N-1) \\ x(l-1) & x(l) & \ddots & \vdots \\ \vdots & \ddots & \ddots & x(l+1) \\ x(l-N+1) & \cdots & x(l-1) & x(l) \end{bmatrix}.$$

The matched filtering operation can be represented discretely as

$$\begin{aligned} x_{MF}(l) &= \mathbf{s}^H \mathbf{y}(l) \\ &= \mathbf{s}^H \mathbf{A}(l)\mathbf{s} + \mathbf{s}^H \mathbf{v}(l). \end{aligned} \quad (3)$$

The Minimum Mean Square Error (MMSE) formulation of pulse compression seeks to minimize the cost function

$$\xi(l) = E \left[|x(l) - \mathbf{w}^H \mathbf{y}(l)|^2 \right] \quad (4)$$

for each range cell $l = 0, 1, \dots, L-1$. The solution is a Wiener filter at each range bin,

$$\begin{aligned} \mathbf{w} &= \left(E[\mathbf{y}(l)\mathbf{y}^H(l)] \right)^{-1} E[\mathbf{y}(l)x^*(l)] \\ &= \mathbf{R}_{yy}^{-1} \mathbf{r}_{yx}. \end{aligned} \quad (5)$$

After some algebraic manipulation and assuming that the range profile impulse response is uncorrelated with the noise, the Wiener solution can be written as

$$\mathbf{w} = \rho(l)(\mathbf{C}(l) + \mathbf{R}_{vv})^{-1} \mathbf{s} \quad (6)$$

where $\rho(l) = |x(l)|^2$ and $\mathbf{R}_{vv} = E[\mathbf{v}(l)\mathbf{v}^H(l)]$ is the $N \times N$ noise covariance matrix. The $N \times N$ matrix $\mathbf{C}(l)$ is

$$\mathbf{C}(l) = \sum_{n=-N+1}^{N-1} \rho(l+n) \mathbf{s}_n \mathbf{s}_n^H \quad (7)$$

where \mathbf{s}_n contains the elements of the waveform vector \mathbf{s} shifted by n samples and zero-filled. For example, $\mathbf{s}_2 = [0 \ 0 \ s_0 \ \dots \ s_{N-3}]^T$ and $\mathbf{s}_{-2} = [s_2 \ \dots \ s_{N-1} \ 0 \ 0]^T$.

The MMSE algorithm proceeds iteratively by alternating between estimating the range profile impulse response and the receive filters. To initialize the algorithm, the initial MMSE filter estimate

$$\mathbf{w}_0 = \left(\sum_{n=-N+1}^{N-1} \mathbf{s}_n \mathbf{s}_n^H \right)^{-1} \mathbf{s} \quad (8)$$

may be used. As the algorithm converges, usually in just a few steps, a unique weight vector is computed from successive data segments and applied to each range bin to create the final compressed output. In the presence of large targets, adaptive pulse compression has the benefit of suppressing pulse compression time sidelobes and uncovering weaker targets nearby. However, inverting the covariance matrix in (6) is a computationally expensive step which hinders the real-time implementation of adaptive pulse compression.

This paper proposes a modification to the adaptive pulse compression algorithm which avoids the matrix inversion step by converting the vector estimation problem at each range cell into an equivalent nested sequence of scalar estimation problems. Many of the theoretical tools used to derive the proposed pulse compression algorithm can be found in [3-5] and are briefly summarized in Section II.

II. MULTISTAGE WIENER FILTER

Reconsider the classical linear estimation problem

$$\varepsilon_0(l) = x(l) - x'(l) = x(l) - \mathbf{w}^H \mathbf{y}(l) \quad (9)$$

where $\varepsilon_0(l)$ is the estimation error at delay index l , $x(l)$ is the desired signal, $x'(l)$ is its estimate, \mathbf{w} is the impulse response of the estimation filter and $\mathbf{y}(l)$ is the input data vector. For stationary processes, the optimal linear filter which minimizes the Mean Square Error (MSE), $E[|\varepsilon_0(l)|^2]$, is the Wiener filter

$$\mathbf{w} = \mathbf{R}_{yy}^{-1} \mathbf{r}_{yx} \quad (10)$$

where the covariance matrix \mathbf{R}_{yy} and cross-correlation vector \mathbf{r}_{yx} are

$$\mathbf{R}_{yy} = E[\mathbf{y}(l) \mathbf{y}^H(l)], \quad \mathbf{r}_{yx} = E[\mathbf{y}(l) x^*(l)]. \quad (11)$$

The minimum mean-square error (MMSE) attained with the Wiener filter is

$$\xi_0(l) = \min E[|\varepsilon_0(l)|^2] = \sigma_x^2 - \mathbf{r}_{yx}^H \mathbf{R}_{yy}^{-1} \mathbf{r}_{yx} \quad (12)$$

where $\sigma_x^2 = E[|x(l)|^2]$.

Preprocessing the input data $\mathbf{y}(l)$ by a nonsingular blocking matrix \mathbf{T} prior to Wiener filtering will not change the MMSE. So consider a unitary blocking matrix with the structure

$$\mathbf{T} = \begin{bmatrix} \mathbf{h}_1^H \\ \mathbf{B}_1 \end{bmatrix} \quad (13)$$

with \mathbf{h}_1 equal to the unit vector

$$\mathbf{h}_1 = \frac{\mathbf{r}_{yx}}{\|\mathbf{r}_{yx}\|}. \quad (14)$$

The matrix \mathbf{B}_1 is an $(N-1)$ -by- N unitary matrix which spans the nullspace of \mathbf{h}_1 such that $\mathbf{B}_1 \mathbf{h}_1 = \mathbf{0}$. The matrix \mathbf{B}_1 can be created from [4]

$$\begin{aligned} [\mathbf{U}, \mathbf{S}, \mathbf{V}] &= \text{svd}(\mathbf{h}^T) \\ \mathbf{B}_1 &= [\mathbf{V}(:, 2 : N)]^T. \end{aligned} \quad (15)$$

A faster but less accurate technique to generate \mathbf{B}_1 is as

$$\begin{aligned} [\mathbf{Q}, \mathbf{R}] &= \text{qr}(\mathbf{h}) \\ \mathbf{B}_1 &= [\mathbf{Q}(:, 2 : N)]^T. \end{aligned} \quad (16)$$

Define a new data vector $\mathbf{z}(l)$ as follows

$$\mathbf{z}(l) = \mathbf{T} \mathbf{y}(l) = \begin{bmatrix} x_1(l) \\ \mathbf{y}_1(l) \end{bmatrix}. \quad (17)$$

A new estimation problem equivalent to (9) with MMSE identical to (12) can be defined using the transformed data vector $\mathbf{z}(l)$ as

$$\varepsilon_0(l) = x(l) - \mathbf{w}_z^H \mathbf{z}(l). \quad (18)$$

The Wiener filter for the transformed input process is equal to

$$\mathbf{w}_z = \mathbf{R}_{zz}^{-1} \mathbf{r}_{zx}. \quad (19)$$

The covariance matrix \mathbf{R}_{zz} , its inverse \mathbf{R}_{zz}^{-1} , and the cross-correlation vector \mathbf{r}_{zx} are

$$\mathbf{R}_{zz} = \begin{bmatrix} \sigma_{x_1}^2 & \mathbf{r}_{y_1 x_1}^H \\ \mathbf{r}_{y_1 x_1} & \mathbf{R}_{y_1 y_1} \end{bmatrix} \quad (20)$$

$$\mathbf{R}_{zz}^{-1} = \frac{1}{\xi_1} \begin{bmatrix} 1 & -\mathbf{r}_{y_1 x_1}^H \mathbf{R}_{y_1 y_1}^{-1} \\ -\mathbf{R}_{y_1 y_1}^{-1} \mathbf{r}_{y_1 x_1} & \mathbf{R}_{y_1 y_1}^{-1} (\xi_1 \mathbf{I} + \mathbf{r}_{y_1 x_1} \mathbf{r}_{y_1 x_1}^H \mathbf{R}_{y_1 y_1}^{-1}) \end{bmatrix}$$

$$\mathbf{r}_{zx} = E[\mathbf{z}(l)x^*(l)] = [\delta_1 \ 0 \ \dots \ 0]^T \quad (22)$$

where in addition

$$\delta_1 = \sqrt{\mathbf{r}_{yx}^H \mathbf{r}_{yx}}, \quad (23)$$

$$\sigma_{x_1}^2 = \mathbf{h}_1^H \mathbf{R}_{yy} \mathbf{h}_1, \quad (24)$$

$$\mathbf{R}_{y_1 y_1} = \mathbf{B}_1 \mathbf{R}_{yy} \mathbf{B}_1^H, \quad (25)$$

$$\mathbf{r}_{y_1 x_1} = \mathbf{B}_1 \mathbf{R}_{yy} \mathbf{h}_1. \quad (26)$$

The partitioned structure of \mathbf{R}_{zz} and \mathbf{R}_{zz}^{-1} suggest that a new $N-1$ dimensional weight vector \mathbf{w}_2 can be defined as

$$\mathbf{w}_2 = \mathbf{R}_{y_1 y_1}^{-1} \mathbf{r}_{y_1 x_1} \quad (27)$$

which is the Wiener filter for estimating the scalar $x_1(l)$ using the vector $\mathbf{y}_1(l)$. The error for the new Wiener filter is given by

$$\varepsilon_1(l) = x_1(l) - \mathbf{w}_2^H \mathbf{y}_1(l) \quad (28)$$

with variance equal to

$$\xi_1 = E[|\varepsilon_1(l)|^2] = \sigma_{x_1}^2 - \mathbf{w}_2^H \mathbf{r}_{y_1 x_1}. \quad (29)$$

A new filtering structure with MMSE identical to ξ_0 is shown in Fig. 1. This structure decomposes the N -dimensional Wiener filter \mathbf{w} into a scalar Wiener filter w_1 and an $(N-1)$ -dimensional vector Wiener filter \mathbf{w}_2 . The reduced dimension vector Wiener filter spans a subspace orthogonal to the subspace spanned by the scalar Wiener filter. The scalar Wiener filter acts on the scalar error signal $\varepsilon_1(l)$ to estimate the scalar desired signal $x(l)$ and is given by

$$w_1 = \frac{\delta_1}{\xi_1}. \quad (30)$$

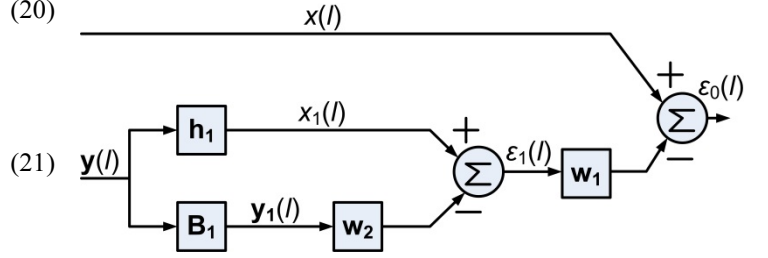


Figure 1. Wiener Filter Decomposition

Repeating this decomposition procedure recursively yields a nested sequence of scalar Wiener filters. At each stage of the recursion the dimension of both the data and the corresponding Wiener filter is reduced by one. The error signal at the output of each stage serves as the scalar observed process for the Wiener filter at the next stage. After $(N-1)$ stages the decomposition is complete and both the data and the Wiener filter have dimension one. An example decomposition is shown in Fig. 2 for the case $N = 4$.

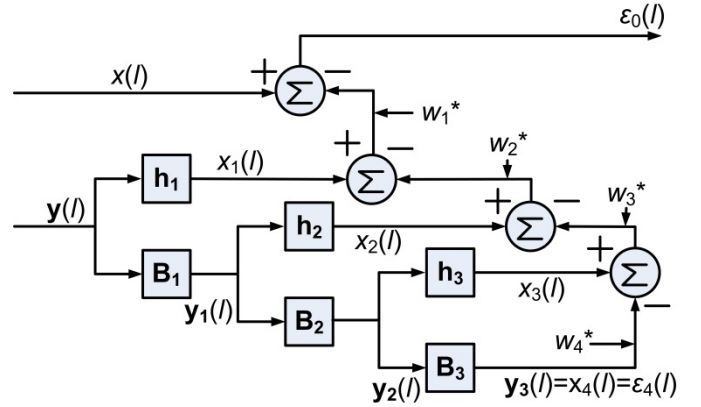


Figure 2. Multistage Wiener Filter With $N = 4$

III. MULTISTAGE WIENER FILTER APPLIED TO ADAPTIVE PULSE COMPRESSION

The MSWF can be applied to the received radar data to yield pulse compressed output identical to the RMMSE filter. The MSWF requires a good estimate of the cross-correlation vector \mathbf{r}_{yx} in (10). Comparing (6) and (10) suggests that

$$\mathbf{r}_{yx} = \rho(l) \mathbf{s}. \quad (31)$$

An initial weight vector \mathbf{w}_0 is computed from (8). Subsequent steps for each iteration of the proposed adaptive pulse compression algorithm proceed as follows;

For $l = 0, \dots, L-1$

1. $x(l) = \mathbf{w}_0^H \mathbf{y}(l)$

2. $\rho(l) = |x(l)|^2$

3. $\mathbf{C}(l) = \sum_{n=-N+1}^{N-1} \rho(l+n) \mathbf{s}_n \mathbf{s}_n^H$
4. $\mathbf{R}_0 = \mathbf{C}(l) + \mathbf{R}_{\mathbf{v}\mathbf{v}}$
5. $\mathbf{r}_0 = \rho(l) \mathbf{s}$
6. $\mathbf{y}_0 = \mathbf{y}(l)$

For $k = 1, \dots, N-1$

7. $\delta_k = \sqrt{\mathbf{r}_{k-1}^H \mathbf{r}_{k-1}}$
8. $\mathbf{h}_k = \frac{\mathbf{r}_{k-1}}{\delta_k}$
9. $x_k(l) = \mathbf{h}_k^H \mathbf{y}_{k-1}$
10. $[\mathbf{U}, \mathbf{S}, \mathbf{V}] = \text{svd}(\mathbf{h}_k^T)$
11. $\mathbf{B}_k = \mathbf{V}(1:N-k+1, 2:N-k+1)^T$
12. $\sigma_{x_k}^2 = \mathbf{h}_k^H \mathbf{R}_{k-1} \mathbf{h}_k$
13. $\mathbf{r}_k = \mathbf{B}_k \mathbf{R}_{k-1} \mathbf{h}_k$
14. $\mathbf{R}_k = \mathbf{B}_k \mathbf{R}_{k-1} \mathbf{B}_k^H$
15. $\mathbf{y}_k = \mathbf{B}_k \mathbf{y}_{k-1}$

end k

$$\xi_N = |\mathbf{y}_{N-1}|^2$$

$$\varepsilon_N = \mathbf{y}_{N-1}$$

$$\delta_N = \sqrt{\mathbf{r}_{N-1}^H \mathbf{r}_{N-1}}$$

For $k = N, \dots, 1$

1. $w_k = \frac{\delta_k}{\xi_k}$
2. if $k > 1$; $\xi_{k-1} = \sigma_{x_{k-1}}^2 - w_k \delta_k$

end k

end l

The complete transformation applied to the input signal vector $\mathbf{y}(l)$ to arrive at the pulse compressed sample $x'(l)$ for range bin l is given by

$$x'(l) = \mathbf{w}_{MSWF}^H \mathbf{T} \mathbf{y}(l) \quad (32)$$

where the vector \mathbf{w}_{MSWF} is given by

$$\mathbf{w}_{MSWF} = \left[w_1^* \quad -w_1^* w_2^* \quad \dots \quad (-1)^{N+1} \prod_{i=1}^N w_i^* \right]^T \quad (33)$$

and the matrix \mathbf{T} is

$$\mathbf{T} = \begin{bmatrix} \mathbf{h}_1^H \\ \mathbf{h}_2^H \mathbf{B}_1 \\ \vdots \\ \mathbf{h}_{N-1}^H \prod_{i=N-2}^1 \mathbf{B}_i \\ \prod_{i=N-1}^1 \mathbf{B}_i \end{bmatrix}. \quad (34)$$

IV. COMPUTATIONAL COMPLEXITY

The bulk of the computational complexity inherent in the RMMSE adaptive pulse compression algorithm resides in inverting the covariance matrix in (6). In practice the $N \times N$ covariance matrix would not be inverted directly however. Instead an LU decomposition of the matrix would be formed and the resulting system of equations would be solved for the optimal weight vector using forward or backward substitution. The total number of computations required is $2N^3/3 + N^2$ [6].

The most computationally intensive step in the MSWF algorithm is computing the orthogonal blocking matrix \mathbf{B}_k . If a QR decomposition is used to compute \mathbf{B}_k then the total number of computations required at each iteration l is approximately

$$\text{FLOPS}_{\text{MSWF}} = 4 \sum_{M=2}^N (M^2 - M). \quad (35)$$

The computational count for the MSWF algorithm can be reduced by pruning the number of stages in the wiener filter decomposition. For example, assume $\mathbf{C}(l) = \mathbf{0}$ and $\mathbf{R}_{\mathbf{v}\mathbf{v}} = \sigma^2 \mathbf{I}$ in (6). Then the cross-correlation vector is collinear with the Wiener filter and only the first stage in the decomposition is necessary to minimize the mean square error.

V. SIMULATED RESULTS

Figure 3 shows some simulated results using the proposed adaptive pulse compression algorithm. The plot shows a single point target at range bin 69 with a 20 dB SNR. The radar waveform is a 32-bit maximal length shift register code. Figure 4 is a close-up view of the region on either side of the target. Both plots show that the RMMSE filter significantly reduces the time sidelobes of the compressed waveform compared to the output of a matched filter (shown in red). The output of the MSWF algorithm (shown in blue) is exactly equal to the RMMSE output (shown in black) and the two curves coincide.

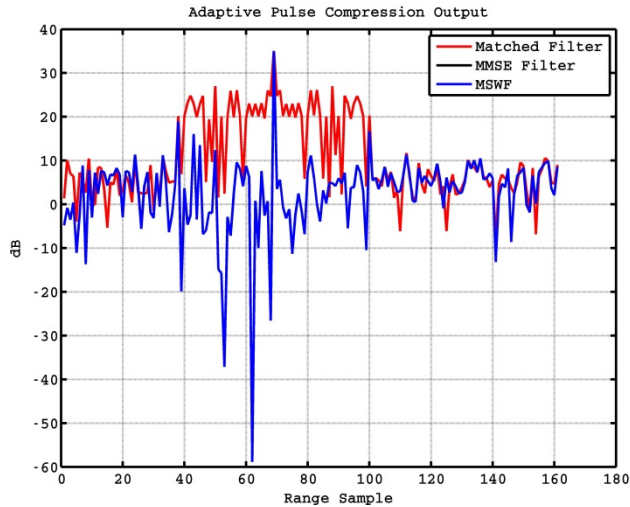


Figure 3. Adaptive Pulse Compression Output

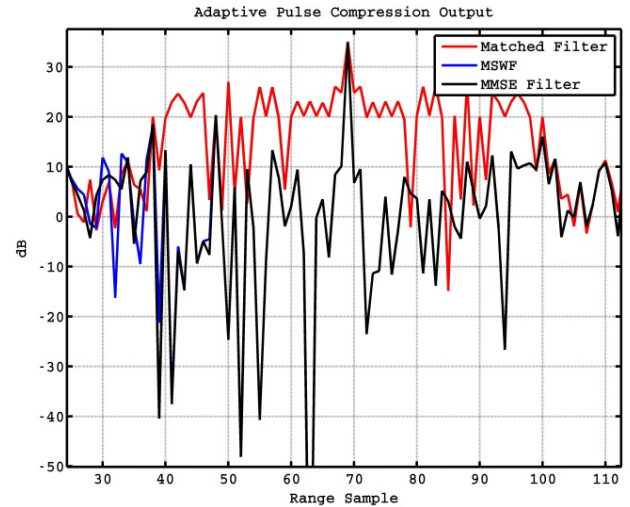


Figure 5. Compressed Output of Reduced Rank Filter

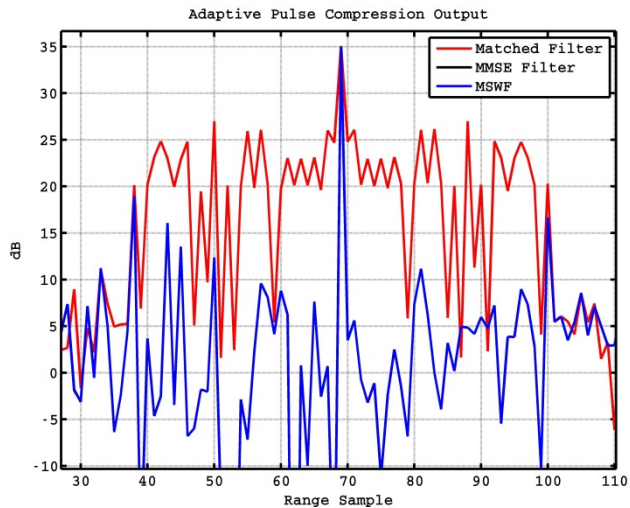


Figure 4. Close-Up View of Target

Figure 5 illustrates the compressed output of the MSWF if only the first 23 stages in the decomposition are retained. As can be seen in the plot there are some small errors that appear at the output of the reduced rank filter.

VI. CONCLUSIONS

This paper describes a new technique for adaptive pulse compression based on a multi-stage decomposition of the Wiener filter. This decomposition projects the input data vector onto a nested sequence of orthogonal subspaces and applies a scalar Wiener filter to the output of each stage. The final result is identical to the standard RMMSE output. Furthermore, a reduced rank processor can be created with lower computational complexity by pruning the number of stages utilized in the decomposition.

REFERENCES

- [1] S. Blunt and K. Gerlach, "Adaptive Pulse Compression via MMSE Estimation," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 42, No. 2, April 2006.
- [2] S. Blunt and T. Higgins, "Dimensionality Reduction Techniques for Efficient Adaptive Pulse Compression," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 46, No. 1, January 2010.
- [3] J. S. Goldstein, "A Multistage Representation of the Wiener Filter Based on Orthogonal Projections," *IEEE Transactions on Information Theory*, Vol. 44, No. 7, November 1998.
- [4] J. S. Goldstein and I. S. Reed, "Theory of Partially Adaptive Radar," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 33, No. 4, October 1997.
- [5] M. L. R. de Campos, S. Werner, and J. A. Apolinario Jr., "On an Efficient Implementation of the Multistage Wiener Filter Through Householder Reflections for DS-CDMA Interference Suppression," *Proceedings 2003 Global Telecommunications Conference*, Rio de Janeiro, Brazil, December 1-5, 2003.
- [6] G. H. Golub and C. F. Van Loan, *Matrix Computations*, 3rd ed., The Johns Hopkins University Press, Baltimore, MD., 1996.