# Adaptive Filtering and its Applications in Satellite Communications

Mohammed Shamma

Analex Corporation

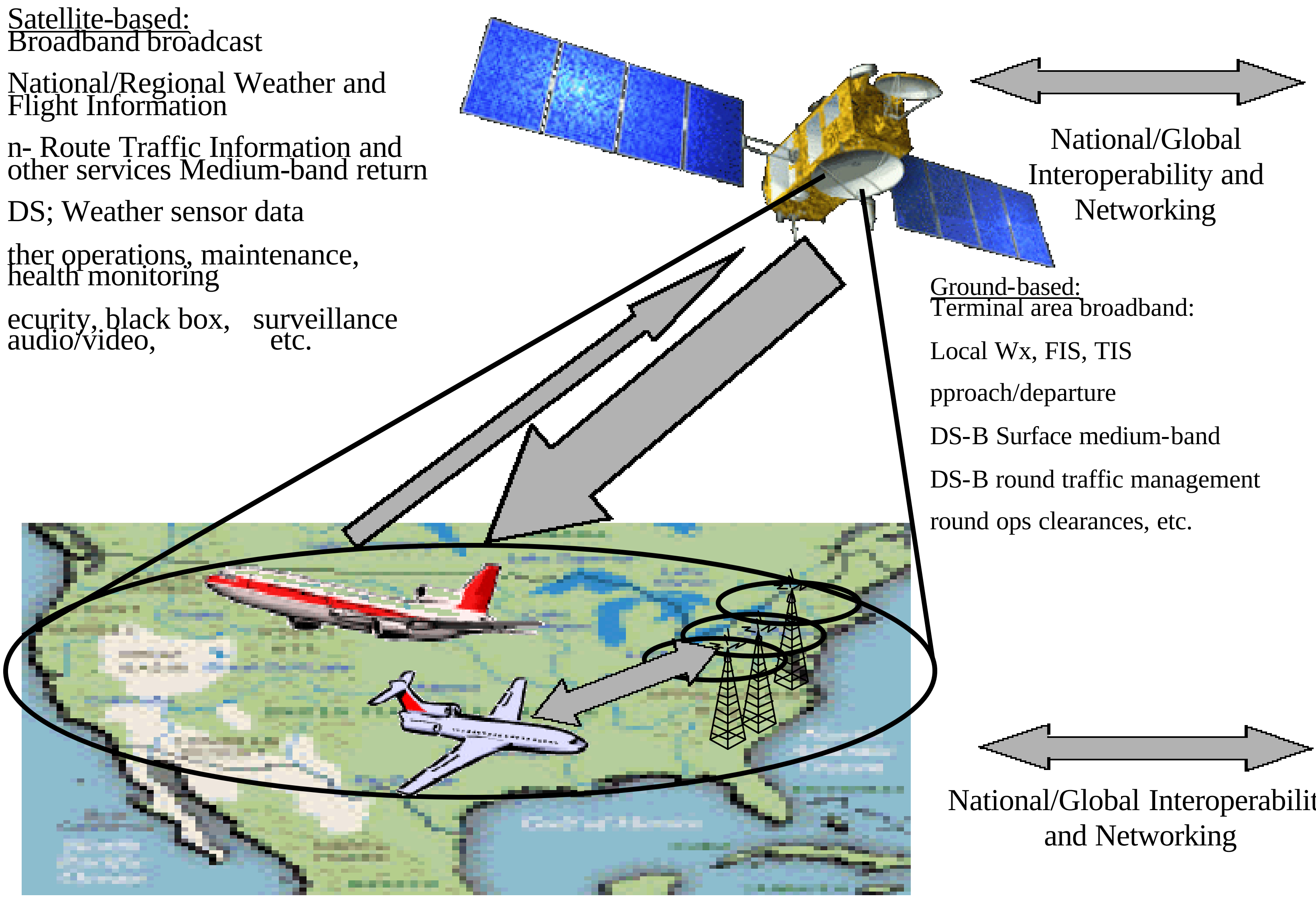
May 2, 2002

# Objectives of Research

- Show various applications of Adaptive Filters within the ATN and AMSS specifically.
- Enhance the theory of Adaptive Filtering by using the Wavelet Transform/LMS scheme instead of only LMS.
- Compare the Wavelet Transform LMS method to other Transform LMS methods such as DCT, DST, ….etc.
- Produce Theoretical and Simulation Results that supports the above objectives and go into the various details and features of the algorithms. Example of such details include time constants or adaptation response, misadjusment, various errors such as dynamic or overall error, stability range, configuration of various implementation methods…etc.



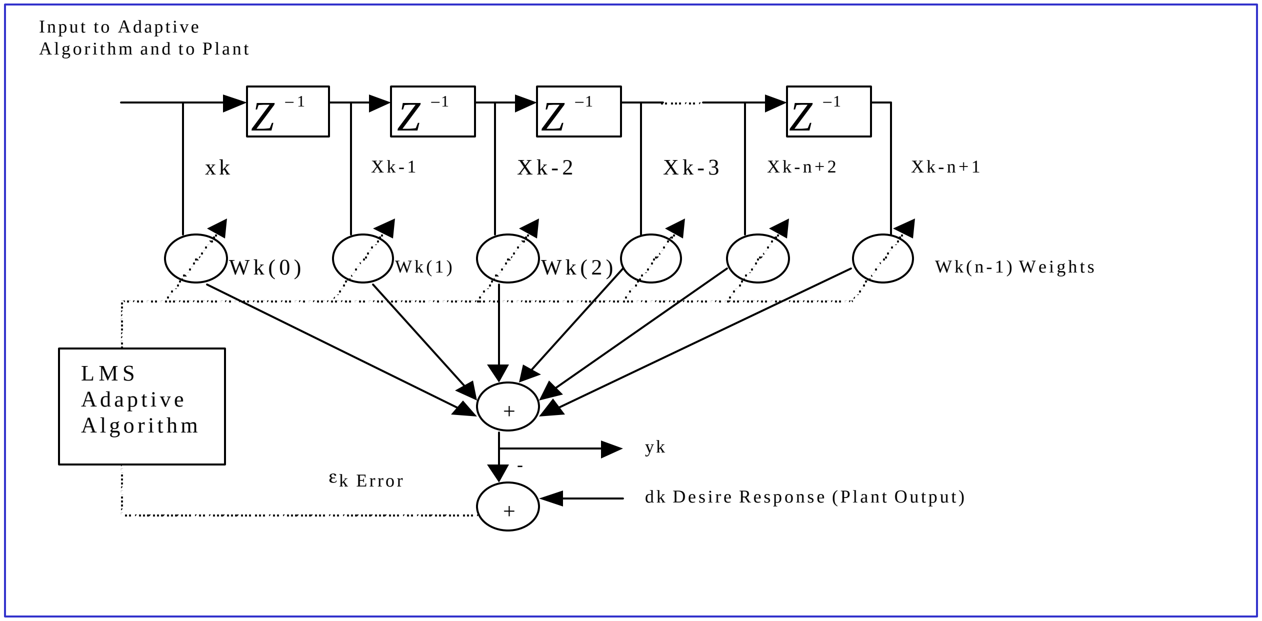
NOTE: Most Results are either shown or referenced in paper


Satellite-based:
Broadband broadcast
National/Regional Weather and Flight Information
n- Route Traffic Information and other services Medium-band return
DS; Weather sensor data
ther operations, maintenance, health monitoring
ecurity, black box, surveillance audio/video, etc.
National/Global Interoperability and Networking
Ground-based:
Terminal area broadband:
Local Wx, FIS, TIS
pproach/departure
DS-B Surface medium-band
DS-B round traffic management
round ops clearances, etc.
National/Global Interoperability and Networking

## What are some of those applications:

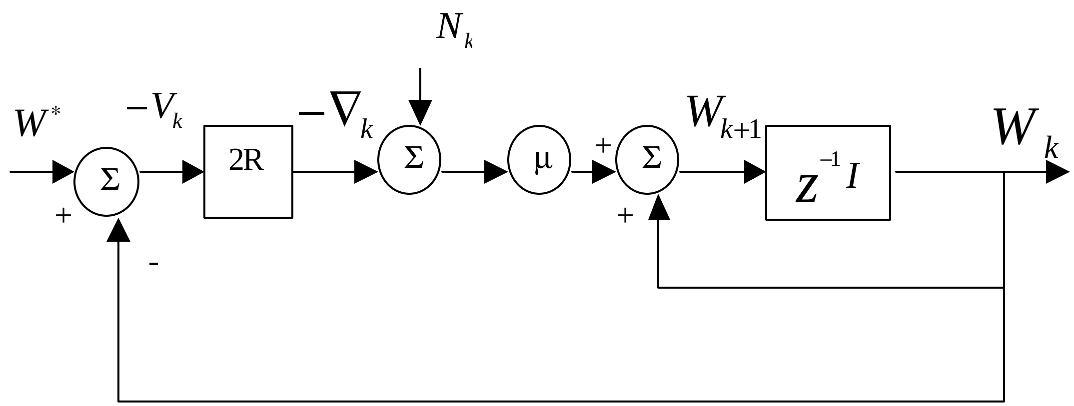
A-Adaptive plant modeling (or channel modeling)
B-Adaptive channel noise canceling in frequency domain (on communication channel) such as narrow band noise in wide band signals and vice versa
C-Adaptive Equalization
E-Adaptive beam forming (space domain)
F-Adaptive voice coding
G-Adaptive rake receivers in CDMA
H-Adaptive cockpit noise canceling (time domain+freq)
I-Adaptive Inverse Control
J-Enhanced GPS receivers … and many more…

# The LMS and Steepest Descent Algorithm




- LMS used in the above to identify a model using an Adaptive Linear Combiner used in many applications in Communications.
- The LMS minimizes the mean square of the error between the desired response and the output of the Adaptive Combiner.

# The LMS and Steepest Descent Algorithm (continued)

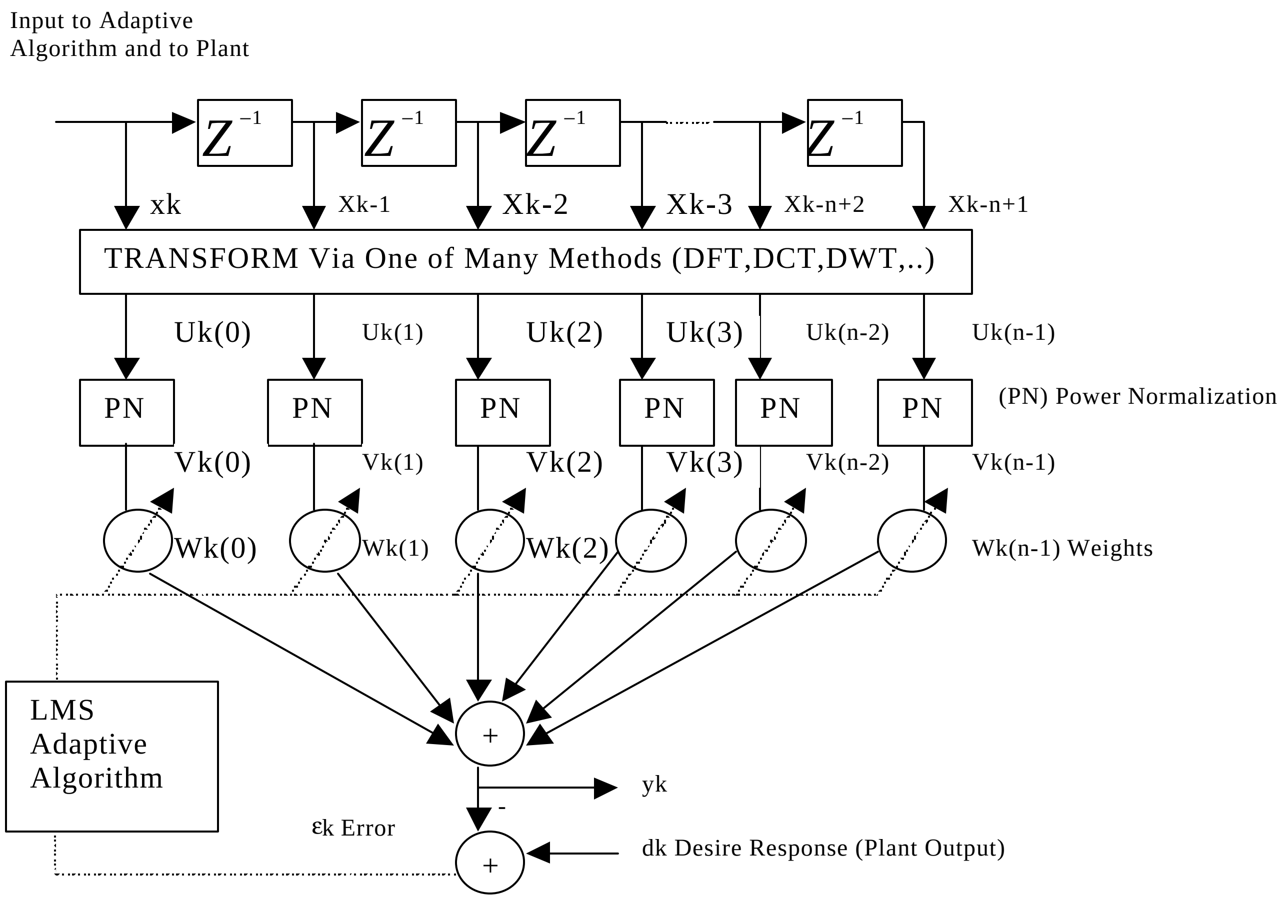



- The LMS is the same as the Steepest Decent method except for the use of an estimate of the gradient as oppose to true gradient $\nabla_k$.
- The LMS converges to the Optimal Wiener solution with some misadjustment and with conditions on the range of the adaptation constant μ.

# LMS Algorithm (continued)

- Misudjustment=$\mu$ trR
- convergence parameter stability range: $0<\mu<1/trR$
- $R=E(X_k X_k^T)$
- trR=sum of eigen values of R $(\lambda_1, \lambda_2, \lambda_3, \lambda_4, \ldots\ldots, \lambda_n)$
- $X_k=[x_{1k}, x_{2k}, x_{3k}, x_{4k}, x_{5k}, x_{6k}, \ldots\ldots\ldots, x_{nk}]^T$
- Learning Curve Time Constant=$\tau_{mse}=1/(4\mu\, \lambda_p)$
- Therefore many of the parameters that the LMS is quantified by such as the speed of adaptation, misadjustment, are dependent on the trace of the autocorrelation of the input signal. It is known that the best results are achieved when the eigenvalue spread of the autocorrelation matrix is the least. This is achieved by transforming the signal as shown next….

# The Transform LMS Algorithm



# Transform LMS Algorithm (continued)

- Transform of input vector

$$S_k = TX_k$$

- Adaptation Weights in transform domain

$$W_{sopt} = (T^t)^{-1} W_{opt}$$

- Error same in both transform and not transform domains

$$e_{s\min} = e_{\min}$$

- Autocorrelation in transform domain

$$R_s = TRT^T$$

- Transform LMS algorithm

$$W_{s_{k+1}} = W_{s_k} + 2m_s S_k e_{s_k}$$

- Adaptation constant bounded by

$$\frac{1}{tr(R_s)} > m_s > 0$$

# Transform LMS Algorithm (continued)

- Self orthogonalizing transform domain LMS algorithm

$$W_{s_{k+1}} = W_{s_k} + 2m_s R_s^{-1} S_k e_{s_k}$$

- Adaptation factor bounded by:

$$\frac{1}{N} = \frac{1}{tr(R_s R_s^{-1})} > m_s > 0$$

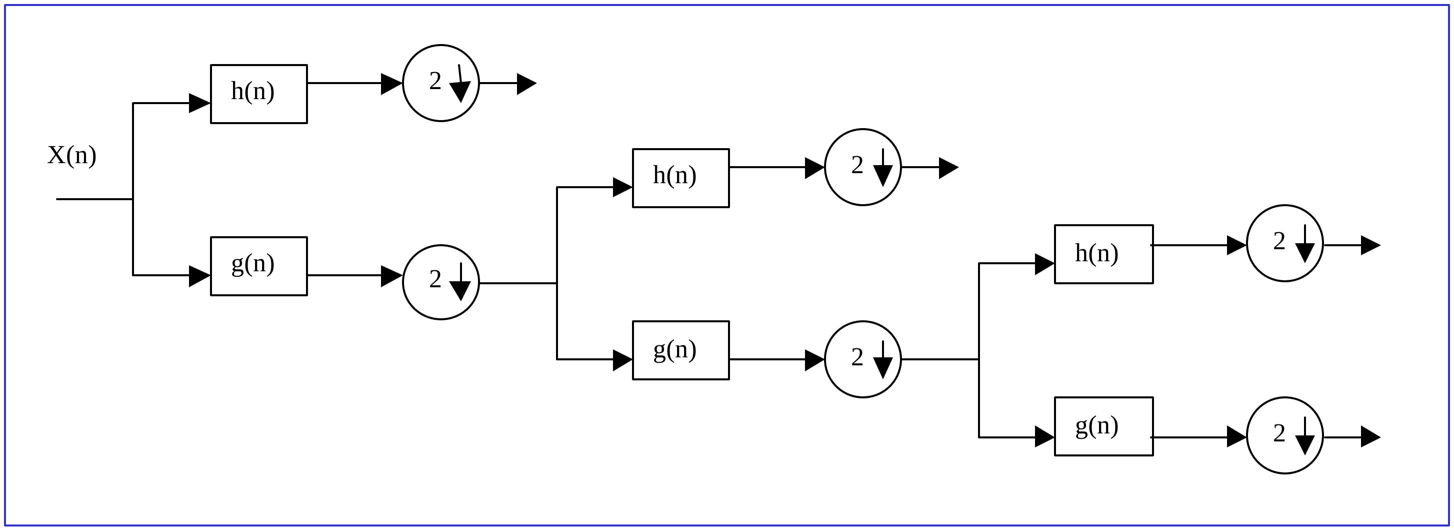
- Recursive algorithm to compute Inverse autocorrelation

$$R_s^n(k,k)^{-1} = b.R_s^{n-1}(k,k)^{-1} + (1-b).Z_s^n(k)^2$$

# Transform LMS Algorithm (continued)

- The advantage of taking the transform of the input signal is to reshape the eigenvalues of the autocorrelation function R. The best solution is to make the maximum over the minimum eigenvalue equal to identity. That is make the spread equal to identity.
- Only few cases where that is possible such as for example if the input is Markov order one then the Discrete Cosine Transform will produce such an optimal result.
- The other case is to use Karhunen Loeve Transform (KLT) but that is dependent on the signal spectrum apriori which makes it an impractical solution.
- The DWT although not necessarily optimal or always better than some of the other methods (such as DCT,DST,DHT, and many more), it is close to optimal in many cases since is does a good job in orthogonalizing the input signal by using Wavelet Transform theory.

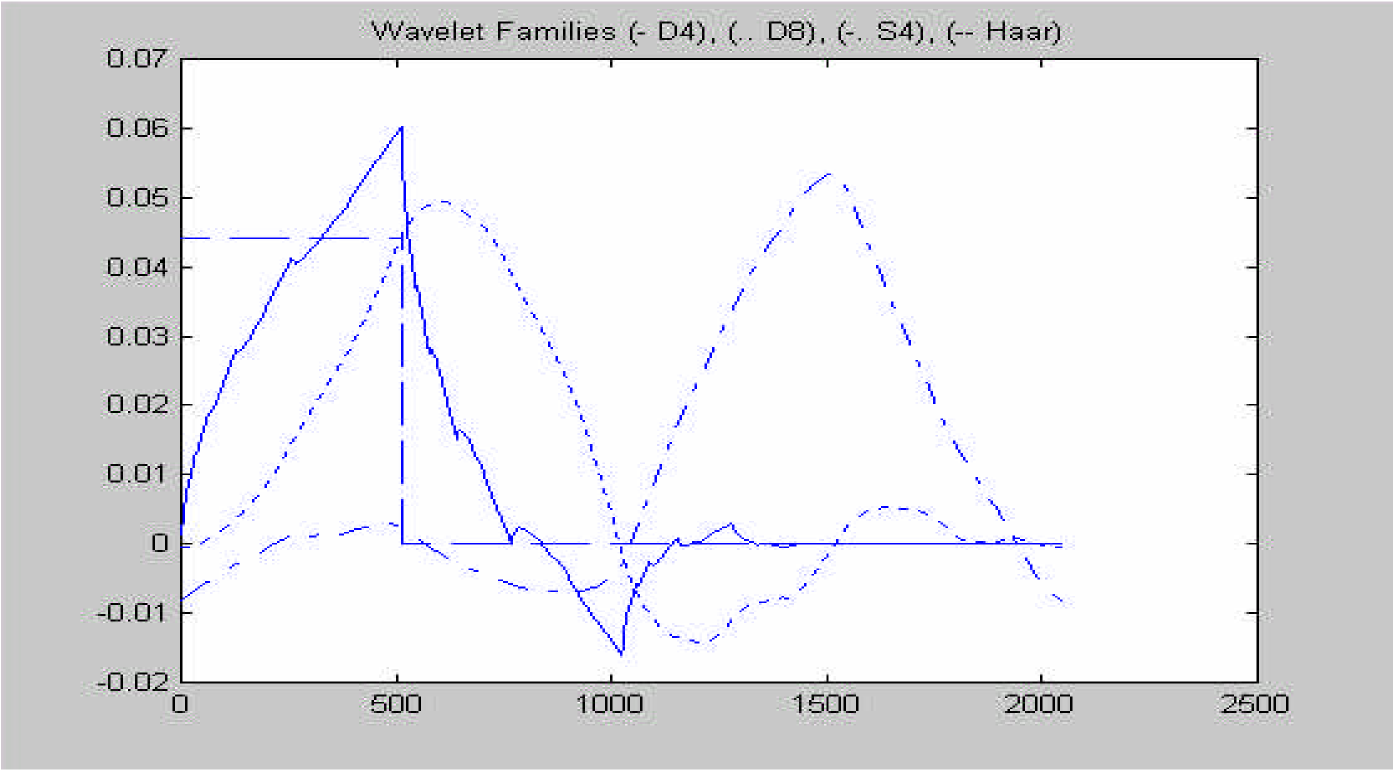
# The DWT




- Theory already developed for the DWT and its implementation.  Uses Subband Coding structure to implement the DWT as shown above.  The difference is in the filters that must satisfy a number of axioms. (Regularity Condition)

# DWT (continued)

- The DWT has many advantages among them is the time and frequency localization feature.
- Various filters exist such as Daubechies, Haar, and others Each highlights different characteristics of the input signal.



# DWT Block Implementation

- size 8x8 with size 4 wavelet coefficient, h0 low pass, h1 high pass

$$T_1 = \begin{bmatrix} h_0(1) & h_0(2) & h_0(3) & h_0(4) & 0 & 0 & 0 & 0 \\ 0 & 0 & h_0(1) & h_0(2) & h_0(3) & h_0(4) & 0 & 0 \\ 0 & 0 & 0 & 0 & h_0(1) & h_0(2) & h_0(3) & h_0(4) \\ h_0(3) & h_0(4) & 0 & 0 & 0 & 0 & h_0(1) & h_0(2) \\ h_1(1) & h_1(2) & h_1(3) & h_1(4) & 0 & 0 & 0 & 0 \\ 0 & 0 & h_1(1) & h_1(2) & h_1(3) & h_1(4) & 0 & 0 \\ 0 & 0 & 0 & 0 & h_1(1) & h_1(2) & h_1(3) & h_1(4) \\ h_1(3) & h_1(4) & 0 & 0 & 0 & 0 & h_1(1) & h_1(2) \end{bmatrix}$$

- Similar structuring applies for any other sizes
- shift by two in each row due to the subsampling by two
- wrapping produces an orthogonal matrix

# DWT Block Implementation (continued)

- subsequent transformations up to highest level look like:

$$T_2 = \begin{bmatrix} h_0(1) & h_0(2) & h_0(3) & h_0(4) & 0 & 0 & 0 & 0 \\ h_0(3) & h_0(4) & h_0(1) & h_0(2) & 0 & 0 & 0 & 0 \\ h_1(1) & h_1(2) & h_1(3) & h_1(4) & 0 & 0 & 0 & 0 \\ h_1(3) & h_1(4) & h_1(1) & h_1(2) & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} T_1(N/2 \times N/2) & 0 \\ 0 & I(N/2 \times N/2) \end{bmatrix}$$

  - N is size of matrix, m depends length of input signal and wavelet filter length

- last level transformation:

$$T_m = \begin{bmatrix} T_1(N/2^m) \times (N/2^m) & 0 \\ 0 & I(N - N/2^m) \times (N - N/2^m) \end{bmatrix}$$

# DWT Block Implementation (continued)

- Last level of Uniform transformation is similarly given by:

$$T_m = \begin{bmatrix} T_1(N/2^m)\times(N/2^m) & 0 \\ 0 & T_1(N/2^m)\times(N/2^m) \end{bmatrix}$$

- the transformed input vector is produced by product of each of the matrices for each level. Hence transformation matrix of the TLMS algorithm becomes:

$$T = T_l \cdots T_3 T_2 T_1$$

# Other Transforms Used

| | |
|---|---|
| Discrete Cosine Transform (DCT) | $T_n(i,l) = C_n(i,l) = \sqrt{2/n}\, K_i \cos\left(\frac{i(l+1/2)p}{n}\right)$ <br> $K_i = 1/\sqrt{2} \quad i = 0$ <br> $K_i = 1 \quad i \neq 0 \qquad \forall i,l = 0 \ldots n-1$ |
| Discrete Fourier Transform (DFT) | $T_n(i,l) = F_n(i,l) = \sqrt{1/n}\, \exp j\left(\frac{2pli}{n}\right)$ <br> $\forall i,l = 0 \ldots n-1$ |
| Discrete Hartley Transform (DHT) | $T_n(i,l) = H_n(i,l) = \sqrt{1/n}\left(\cos\left(\frac{2pli}{n}\right) + \sin\left(\frac{2pi}{n}\right)\right)$ <br> $\forall i,l = 0 \ldots n-1$ |
| Discrete Wavelet Transform (DWT) (already showed) | $h_c(x) = \sum_{n=-\infty}^{\infty} h(n) g_c(2x-n)$ |

# Adaptive Filtering Analyses and Simulation

- Results shown in a theoretical form by computing eigenvalue ratio of maximum to minimum eigenvalues of transform domain input signal for each different transform.

- Results are verified via time domain modeling / identification simulations that employ the TLMS algorithm.

- Various wavelets were tried such as the Daubechies size 4 and 8, Symmlets, and Haar, and other transforms (DFT, DCT,DHT)

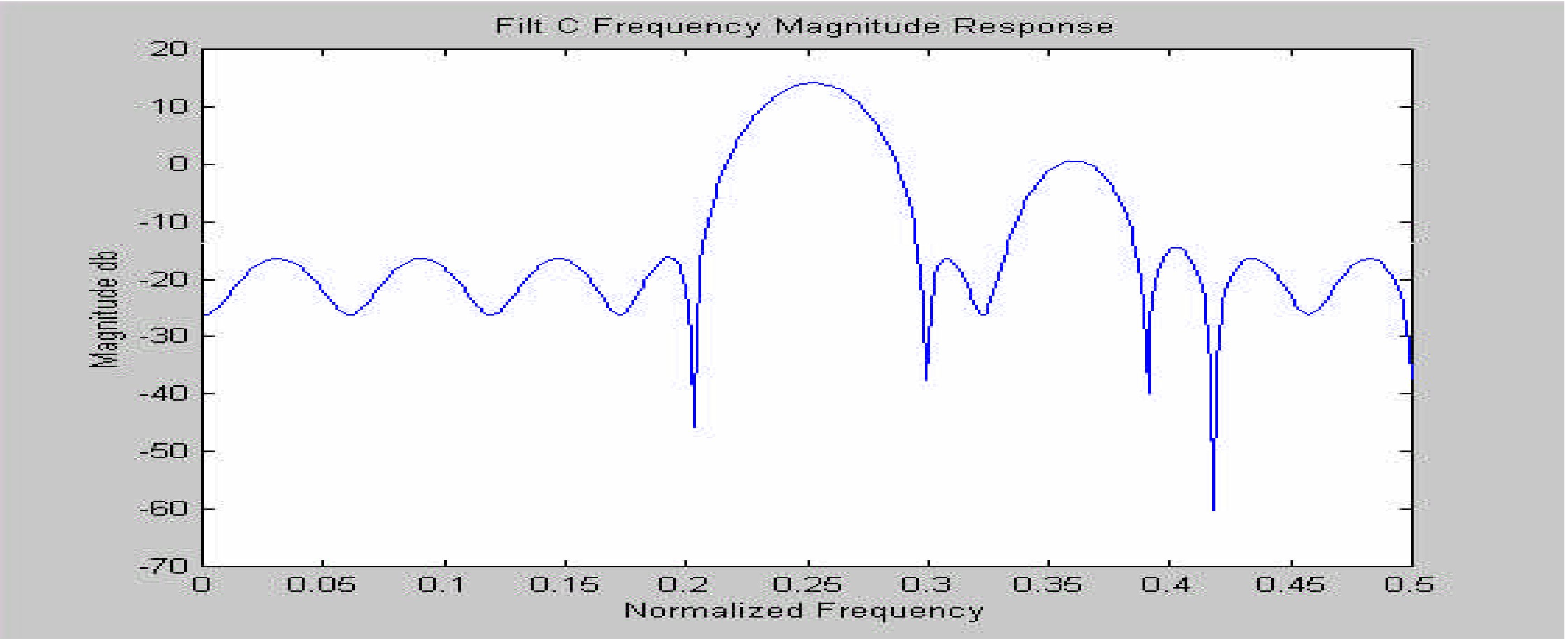
# Adaptive Filtering (continued)

- Many different coloring filters used to test various inputs since that changes the correlation of the transform signal and hence the eigenvalue ratio and the convergence speed

- different length FIR LMS filters were used (size 8 and 16) which changes the size of the block transform matrices

- Simulations are run many times (200 or more) and results are averaged.

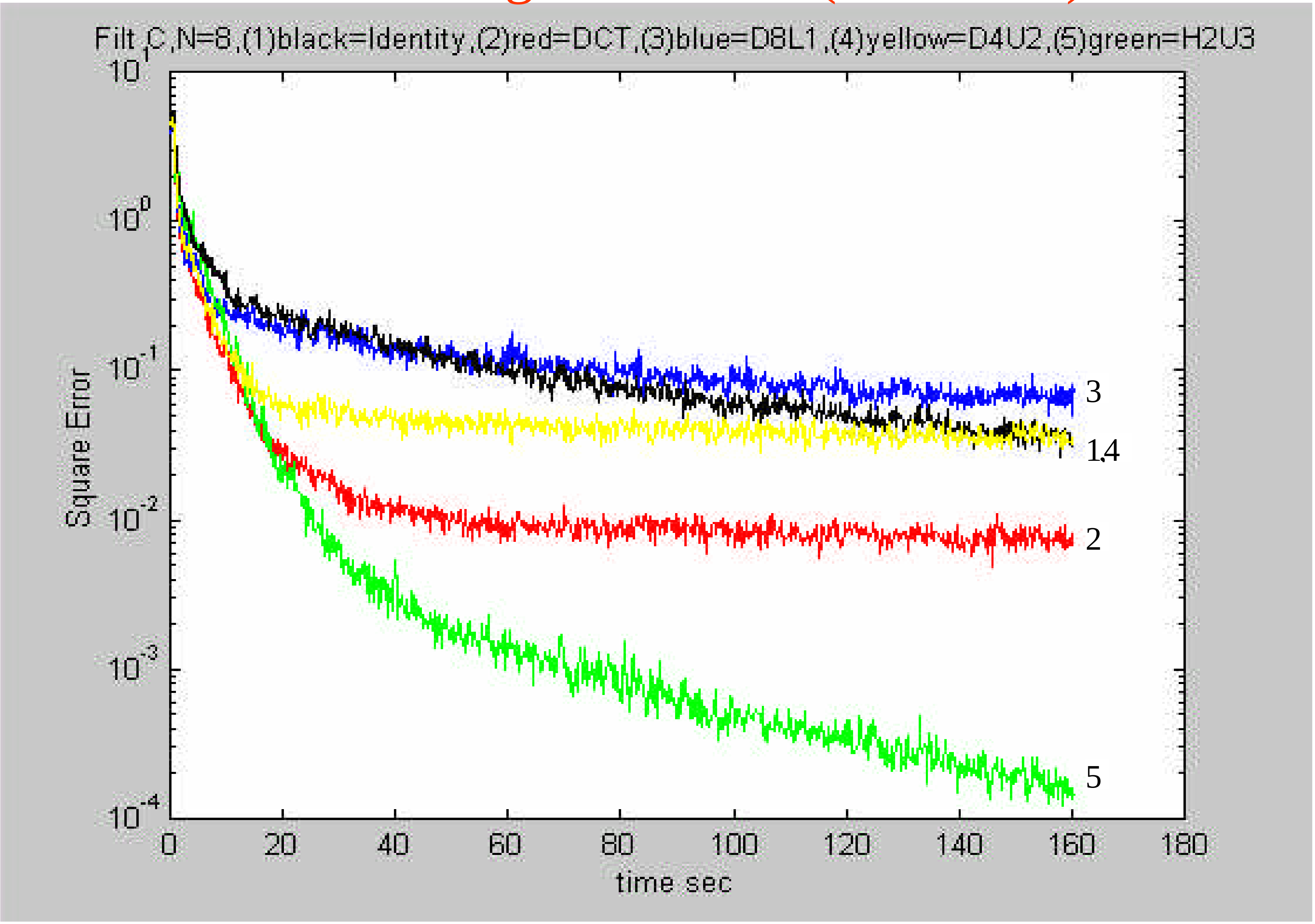
# Example of a Coloring Filter Case



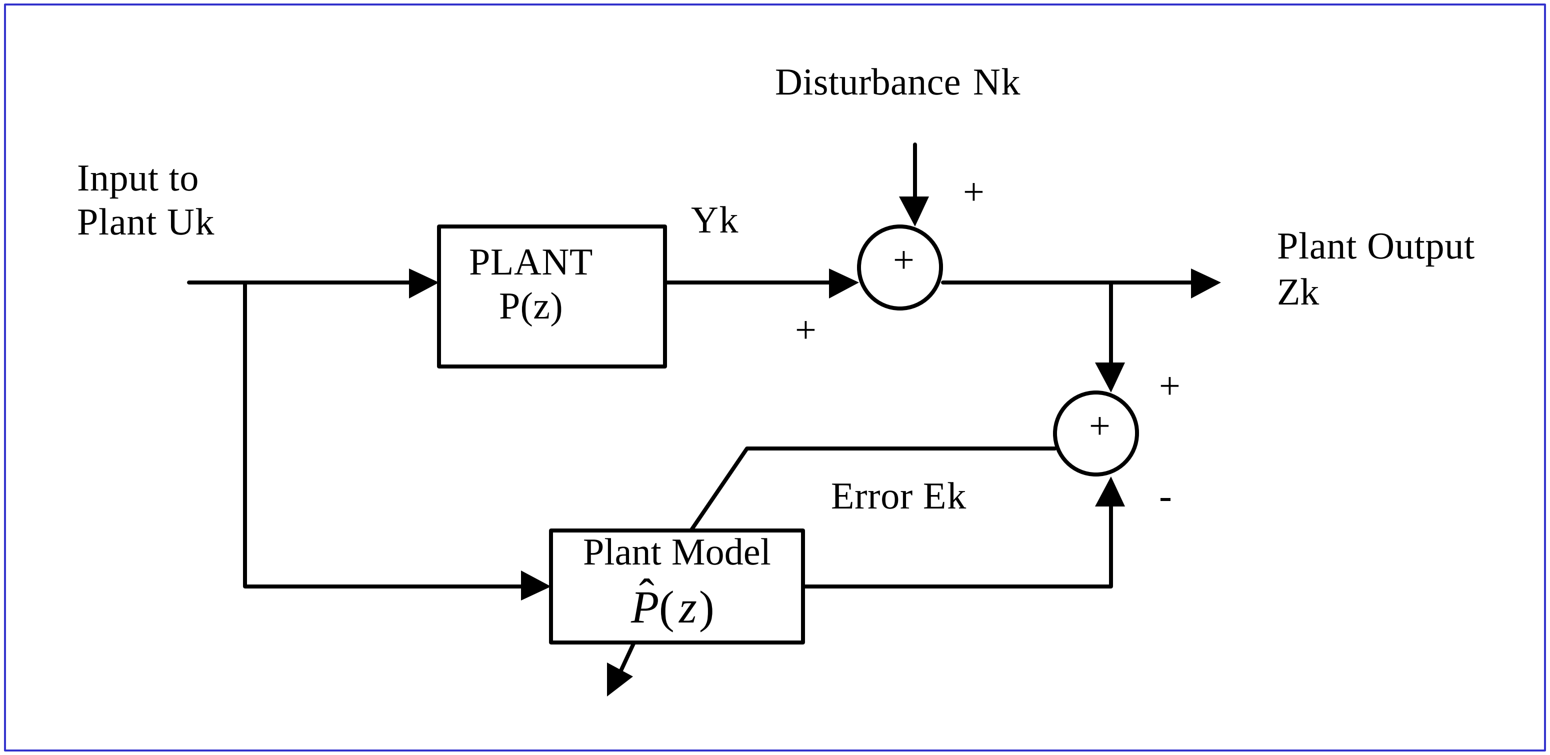

| LEVEL | Uniform/Dyadic (H/L) | DAUBECHIES (8) | DAUBECHIES (4) | HAAR |
|---|---|---|---|---|
| One | ---------------------------- | 629 | 634 | 634 |
| Two | Dyadic Low | ------------------------ | 595 | 416 |
| | Dyadic High | ------------------------ | 628 | 619 |
| | Uniform | ------------------------ | 282 | 151 |
| Three | Dyadic Low | ------------------------ | ------------------------ | 415 |
| | Dyadic High | ------------------------ | ------------------------ | 620 |
| | Uniform | ------------------------ | ------------------------ | 21 |

**Table: Eigen value ratios**
**(EVR=643, DCT=249,DFT=21,DHT=23,PO2=63,WHT=21)**

# First Coloring Filter Case (continued)



# Plant Modeling Block

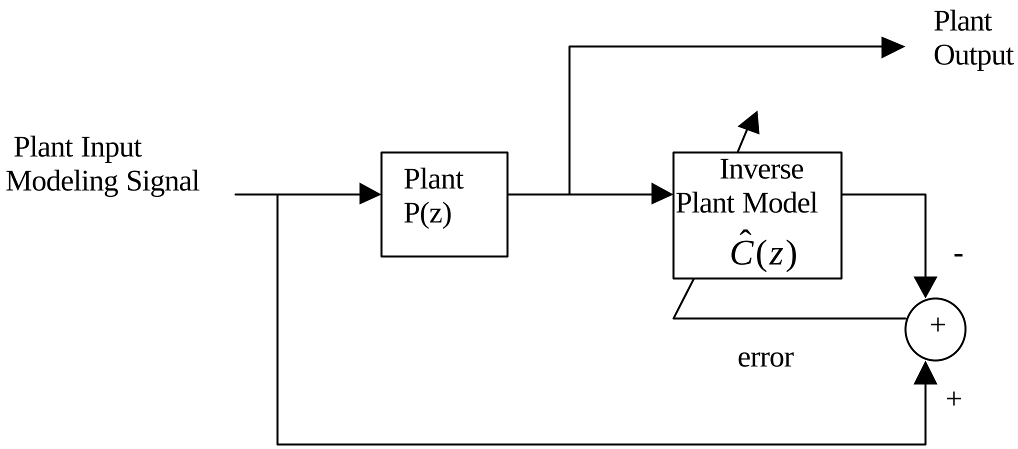



- Identified plant matches plant regardless of noise input

$$\hat{P}^*(z) = P(z)$$

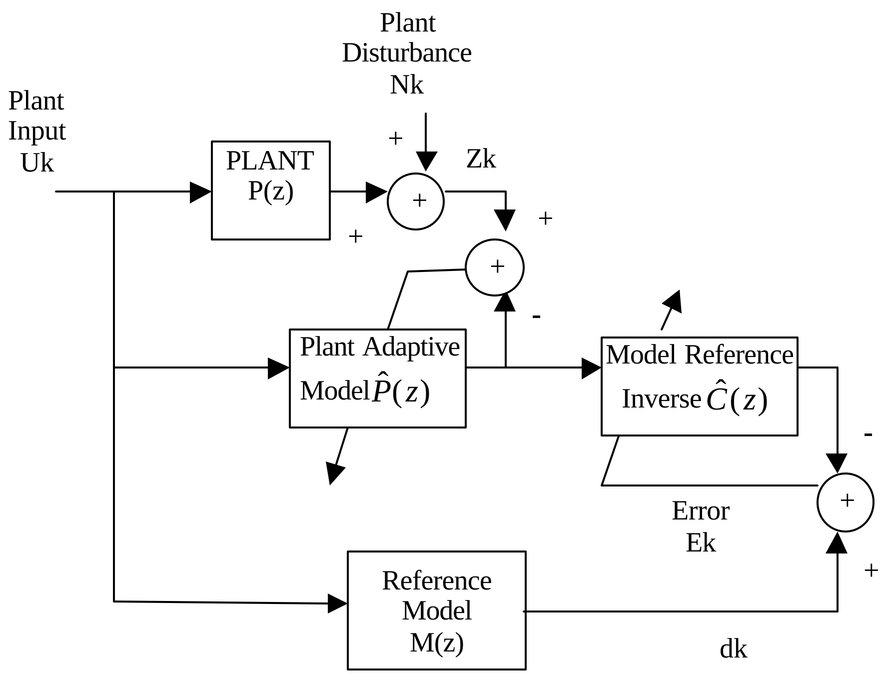
- Identified plant weights (FIR) match those of an IIR plant in order and magnitude given white noise input to plant up to FIR length. "Interestingly similar result is found in DWT/LMS modeling using a different and independent approach"

# Inverse Plant Modeling Block (Widrow)



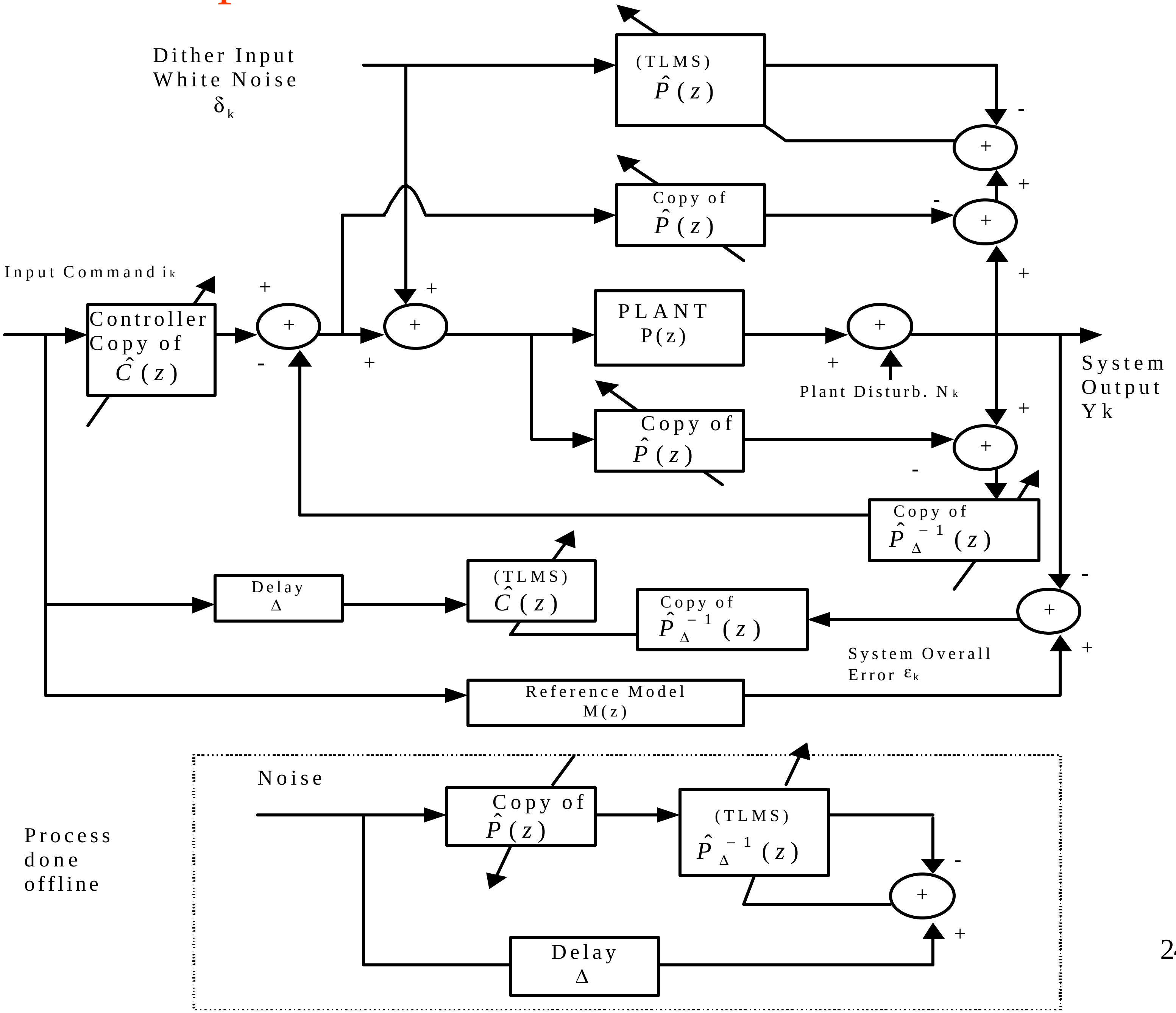

- Minimization of the mean square error produces a perfect inverse:

$$\hat{C}(z) = \frac{P(z^{-1})}{P(z)P(z^{-1})} = \frac{1}{P(z)} = C(z)$$




- Inverse modeling with a reference model and noise on plant. Uses plant model (vs. actual model) to avoid disturbance effects

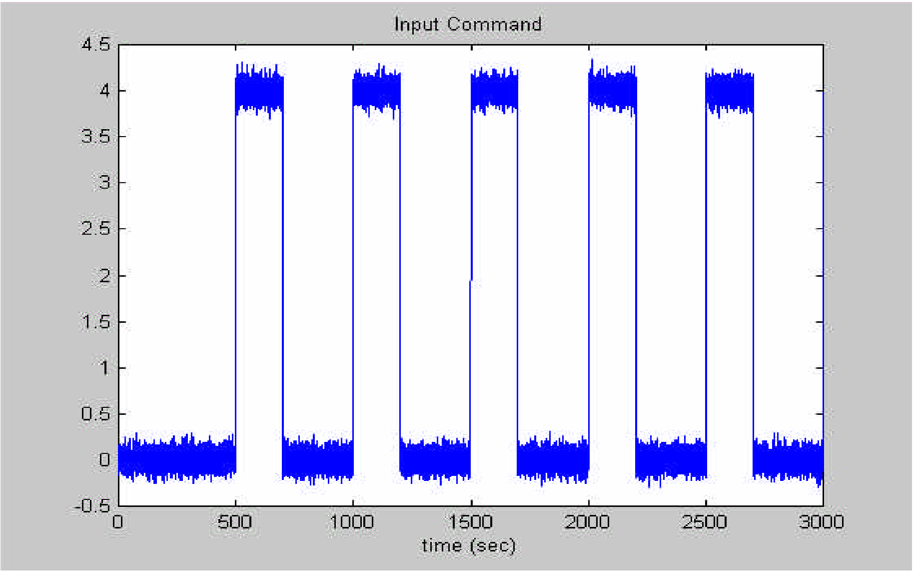
# Transform Adaptive Inverse Controller/Disturbance Canceling

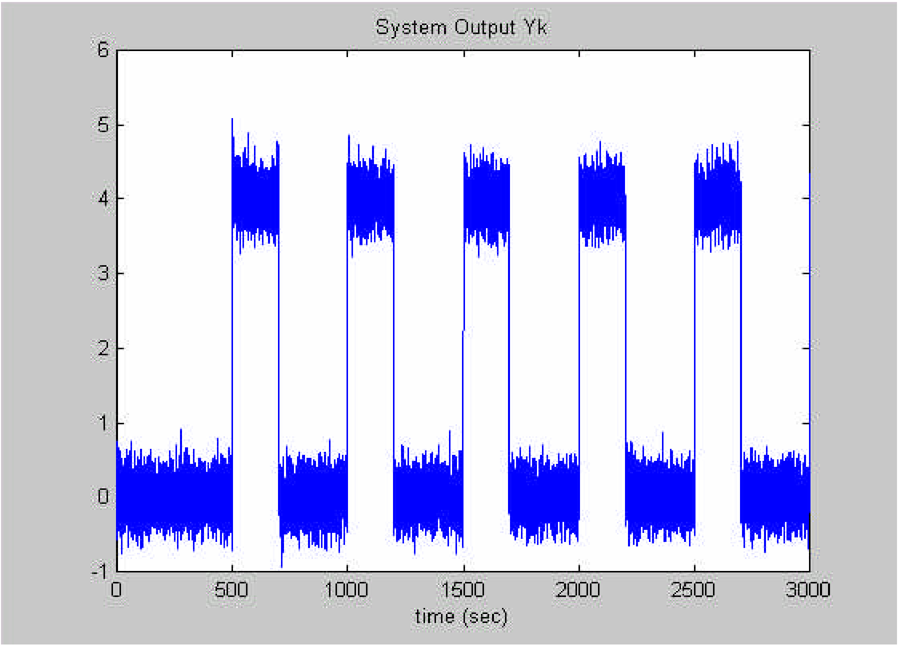


# Simulation Results (A minimum phase plant)

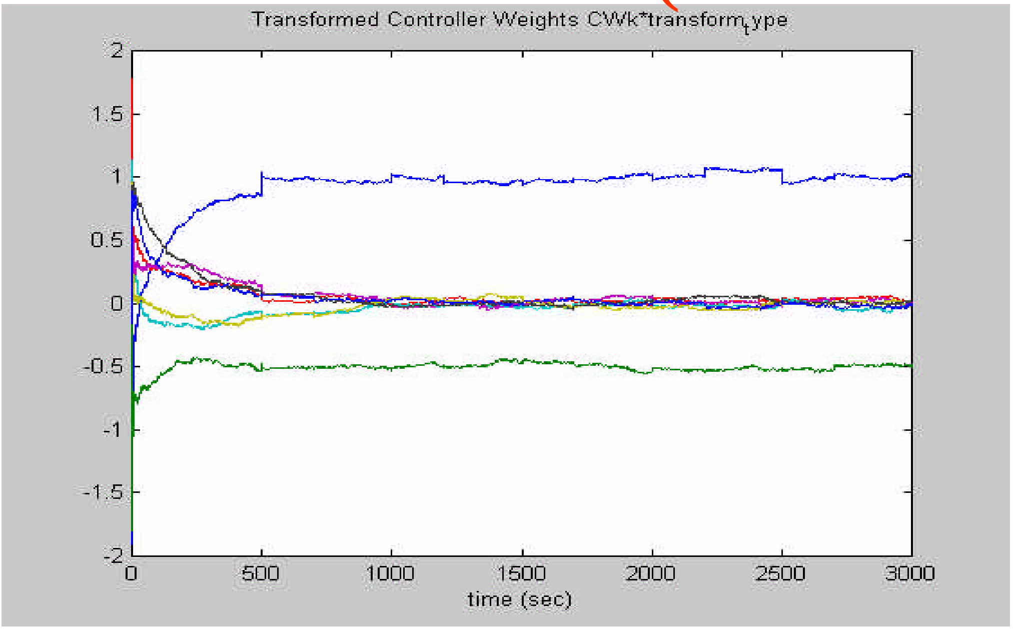
-plant z/(z-0.5), discrete minimum phase.
-adaptation constant=0.0006
-Transform type DWT Haar uniform level 3
-FIR filter length and transform matrix size 8x8
-Reference model unity
-normalization constant 1e-3
-dither noise white, mean of zero, 0.18 std
-weights plotted in the non-transform domain (See TLMS section…). Gives a direct way of checking results (i.e. inverse weights match up with plant denominator -0.5, 1)
-Input command white with 0.08 std. added on a 4 amplitude, 500 steps period pulse train
-Output follows input as expected, weights adapt accordingly

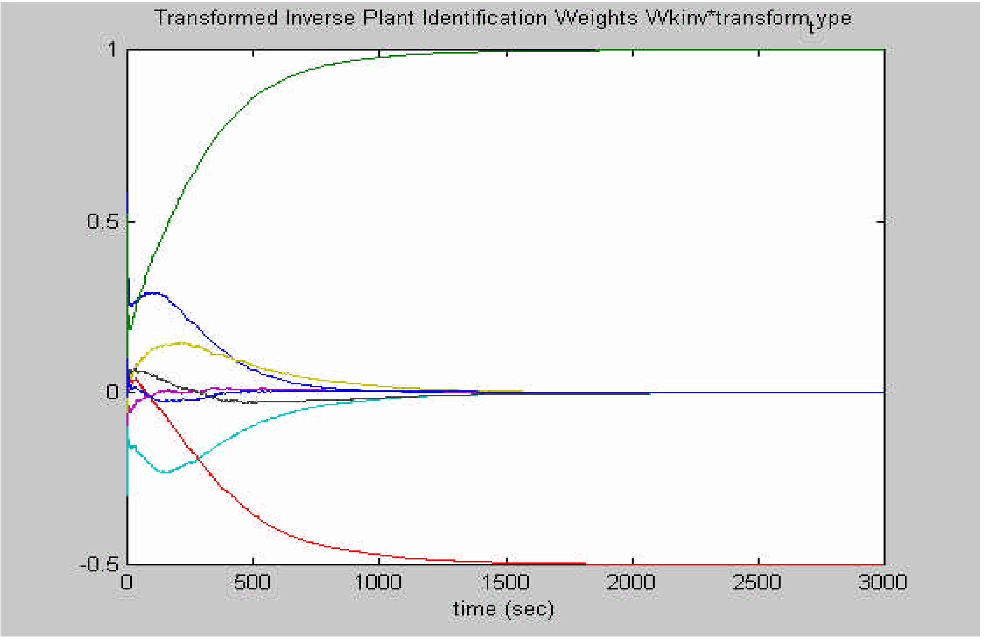
# Simulation Results (continued)



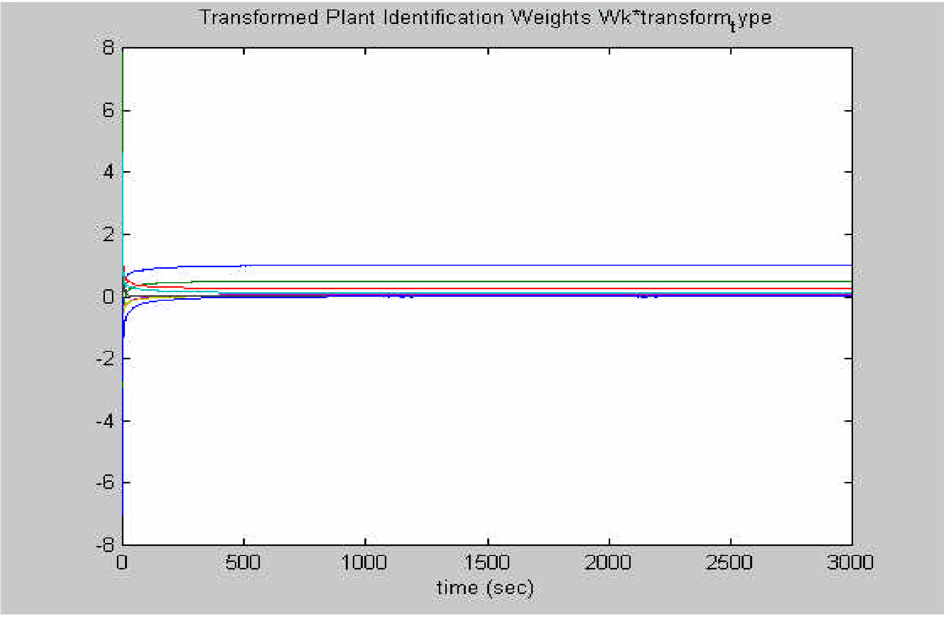



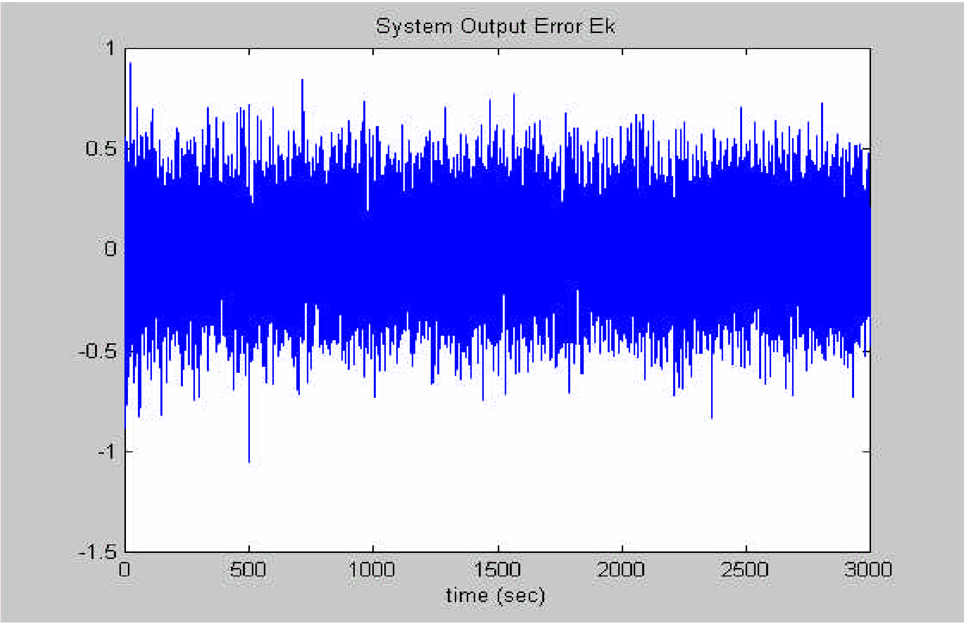
# Simulation Results (continued)



# Simulation Results (continued)



# Simulation Results (Disturbance Canceling-time)

-plant (z)/(z-0.5), discrete, minimum phase.

-adaptation constant 0.0005

-Transform type Haar uniform level 3

-FIR filter length and transform matrix size 8x8

-normalization constant 1e-3

-Reference Model unity

-Disturbance input white, 1 std. Added to amplitude 5 step that starts at 2000 seconds.

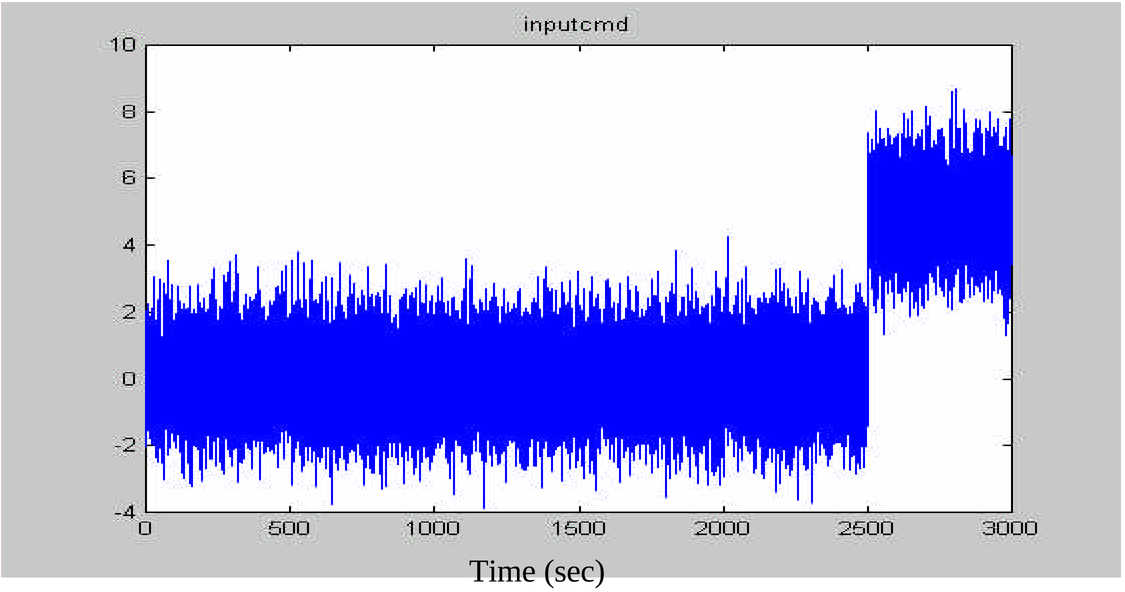
-dither noise white, mean of zero, 1 std

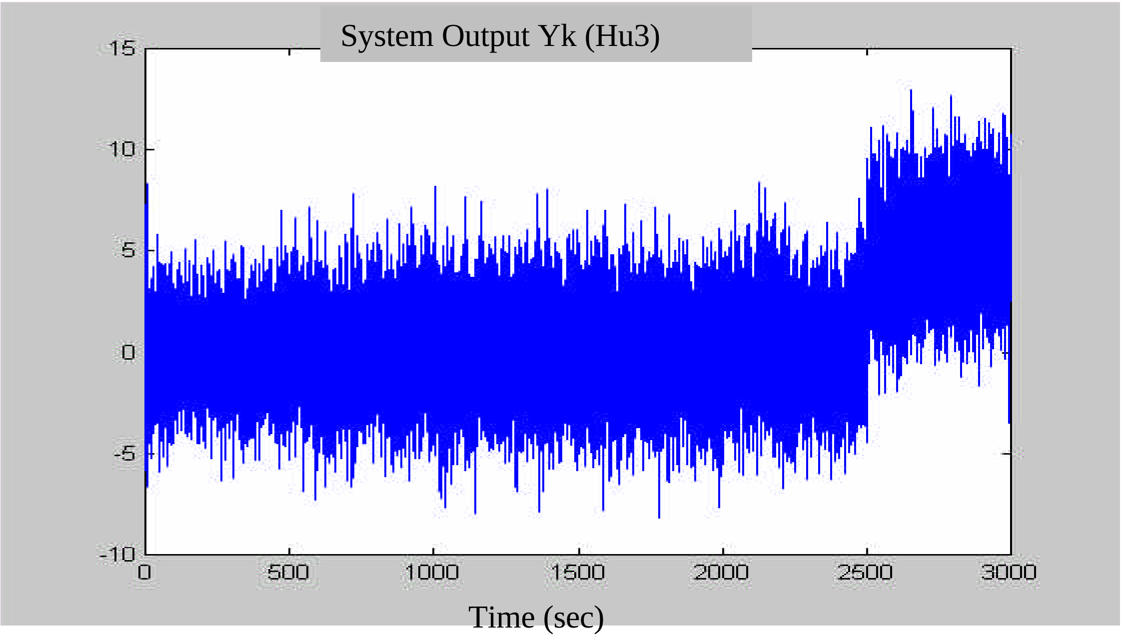
-weights plotted in the non-transform domain (See TLMS section…). Gives a direct way of checking results (i.e. inverse weights match up with plant denominator -0.5, 1)

-Input command white with 1 std. added on a 5 amplitude, step starting at 2500 secs

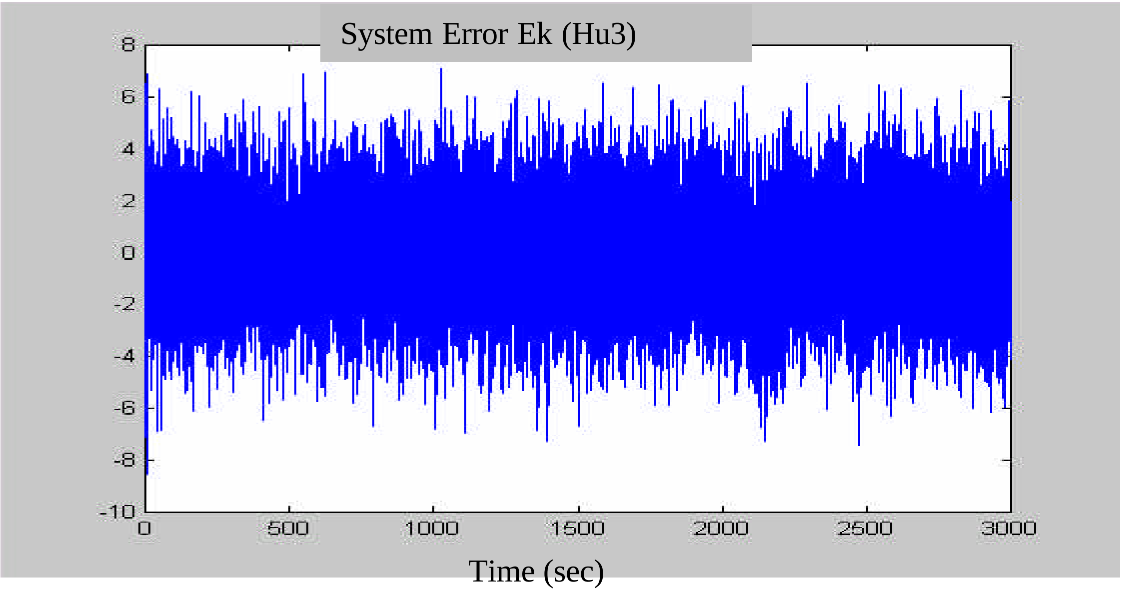
-Output follows input as expected, weights adapt accordingly, disturbance is canceled.

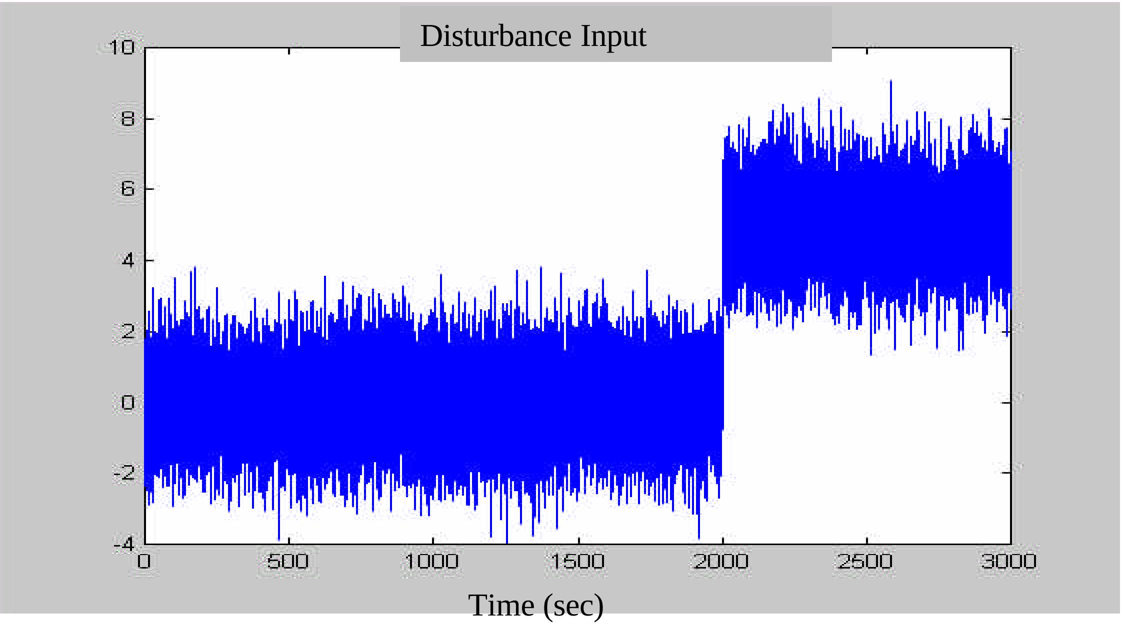
# Simulation Results (continued)



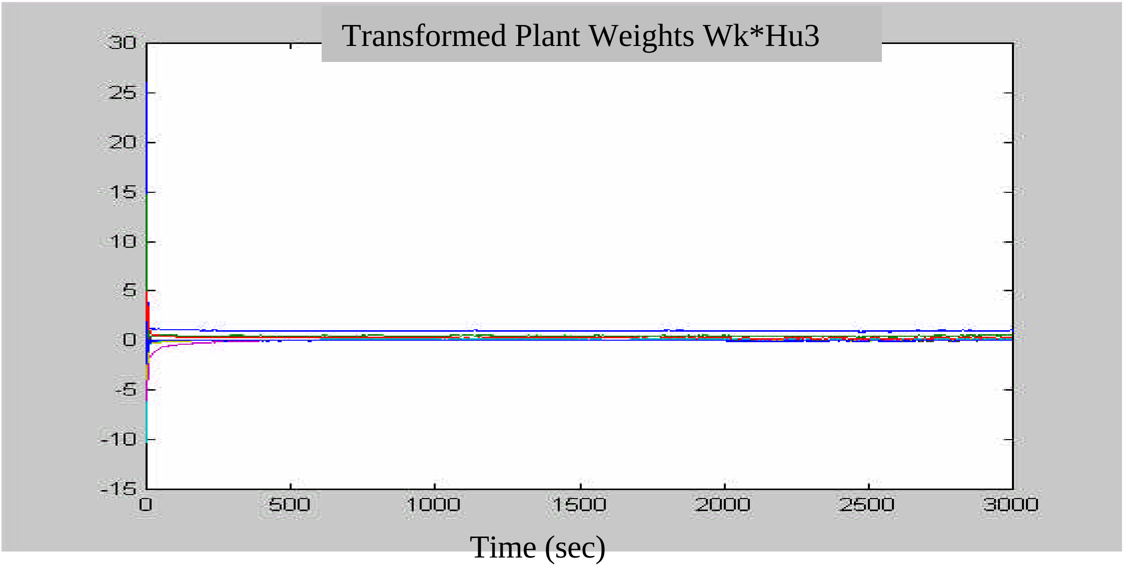



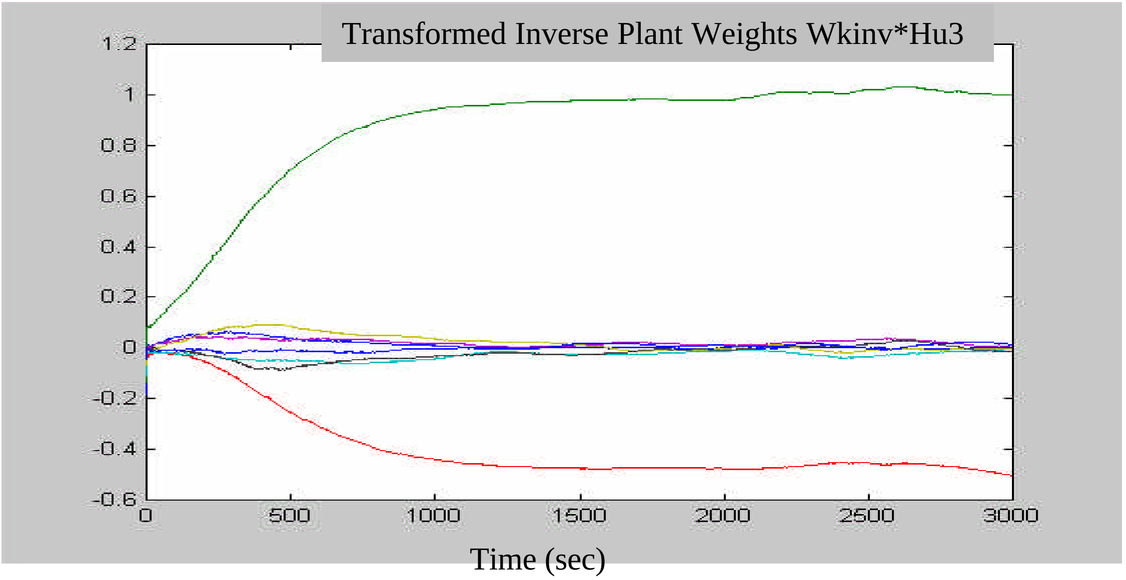
# Simulation Results (continued)



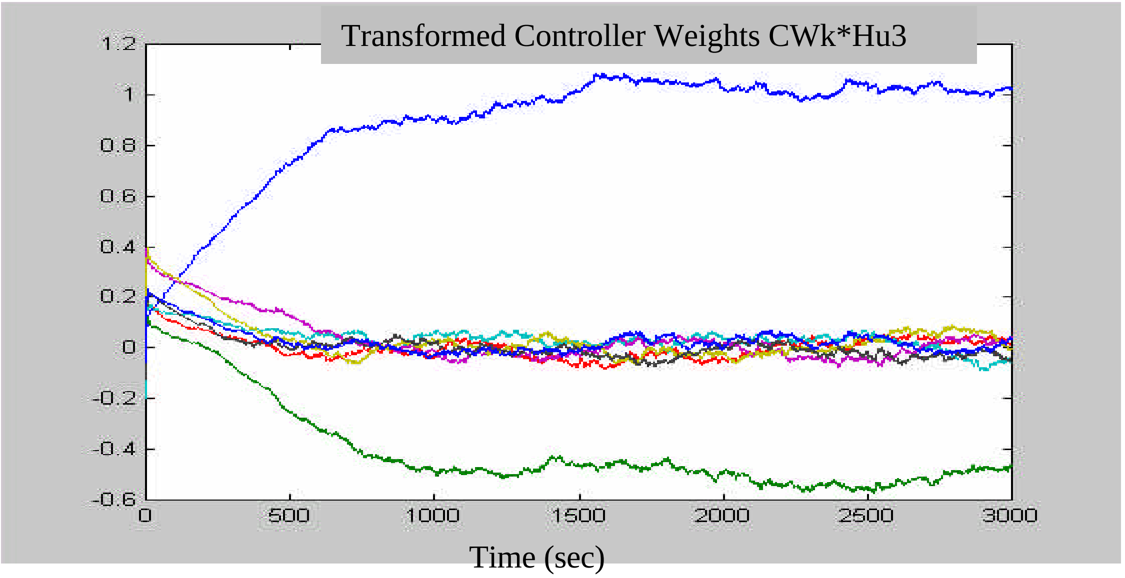



# Simulation Results (continued)

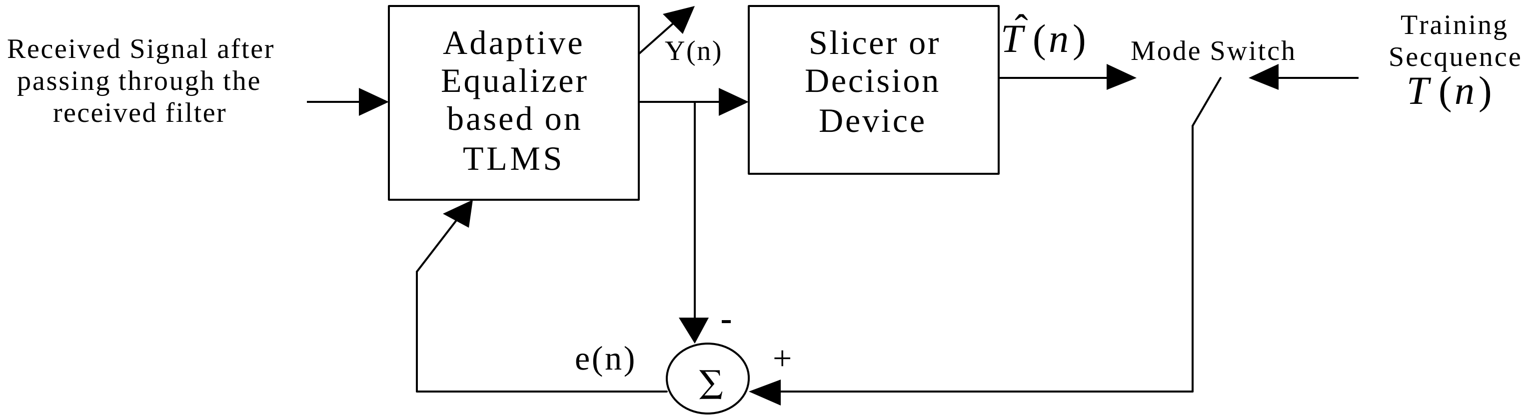


# Simulation Results (continued)



# Adaptive Equalization

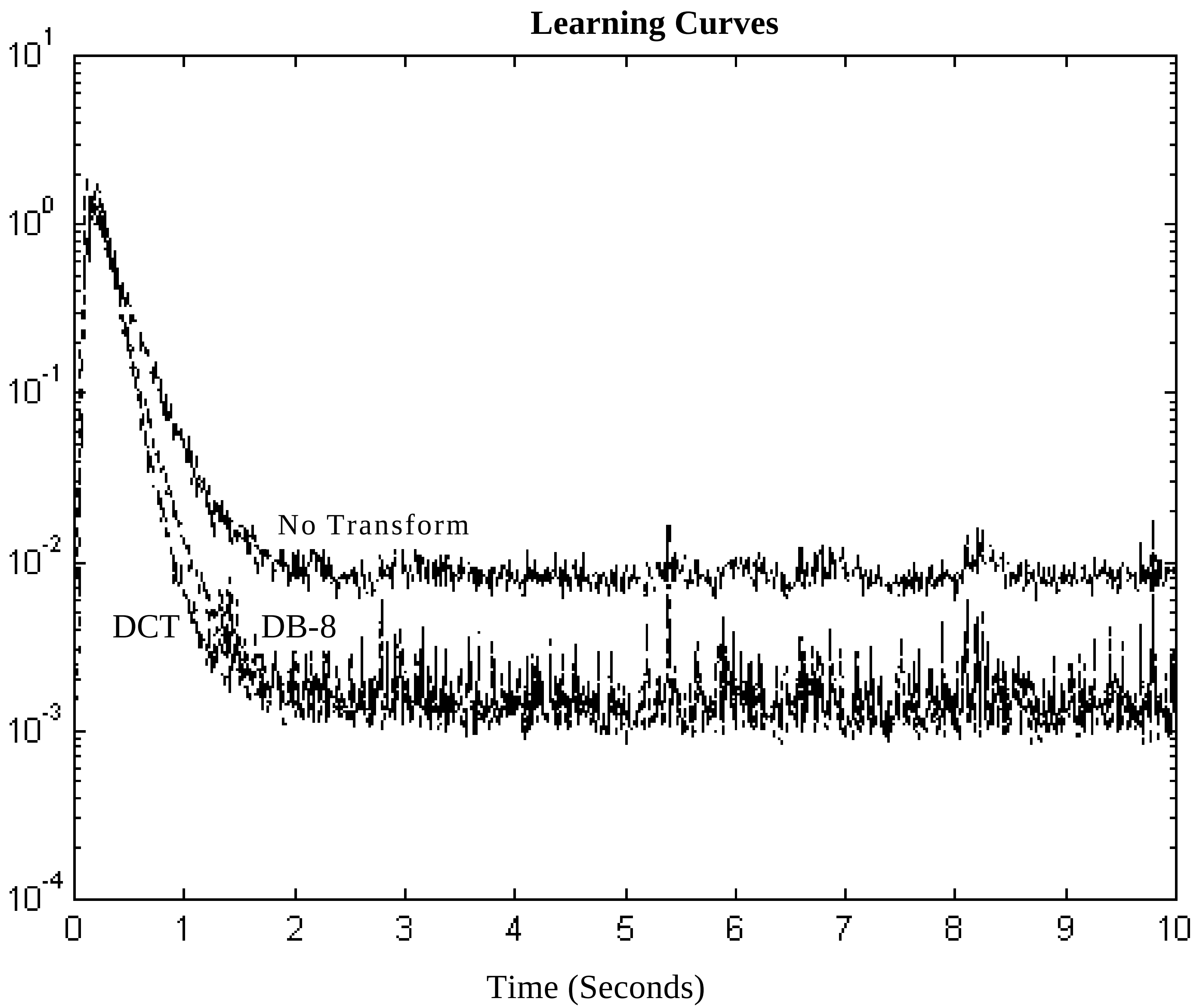



**Adaptive Equalizer**

| Transform Type | Eigenvalue Ratio |
|---|---|
| Haar | 245.5 |
| DCT | 2.9 |
| DFT | 192.3 |
| DHT | 188.3 |
| Daubachies 4 | 159.1 |
| Daubachies 8 | 92.3 |
| Symmlet | 85.6 |
| No Transform | 2320.5 |

**Eigenvalue Ratios of the Transforms used with Pre/Post Filters/Channel model given by Eq:**

$$C_{ch}(z) = 0.05 - 0.063\, z^{-1} + 0.088\, z^{-2} - 0.126\, z^{-3} - 0.25\, z^{-4} + 0.9047\, z^{-5} + 0.25\, z^{-6} + 0.126\, z^{-7} + 0.038 + 0.088\, z^{-8}$$


Learning Curves
10^1
10^0
10^-1
10^-2
10^-3
10^-4
No Transform
DCT
DB-8
0
1
2
3
4
5
6
7
8
9
10
Time (Seconds)

# Noise / Interference canceling (frequency)

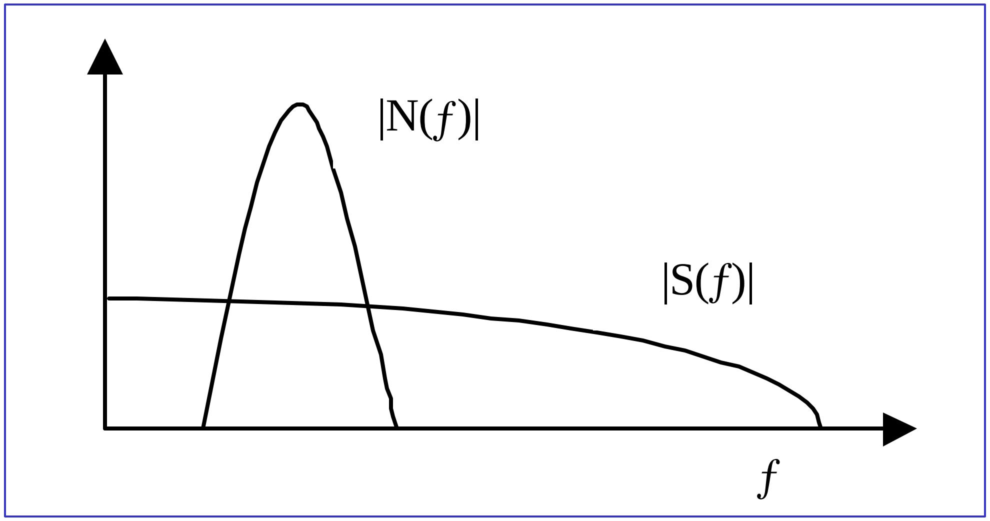



**Figure 7 A strong narrowband interference N(f) in a wideband signal S(f)**

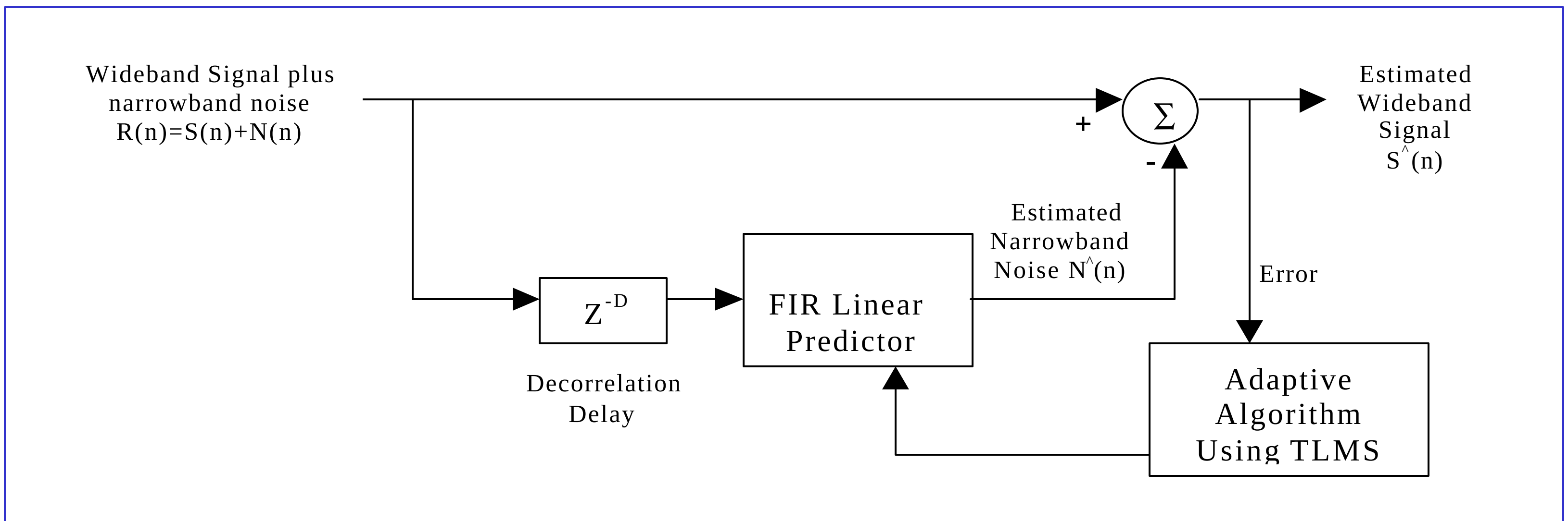
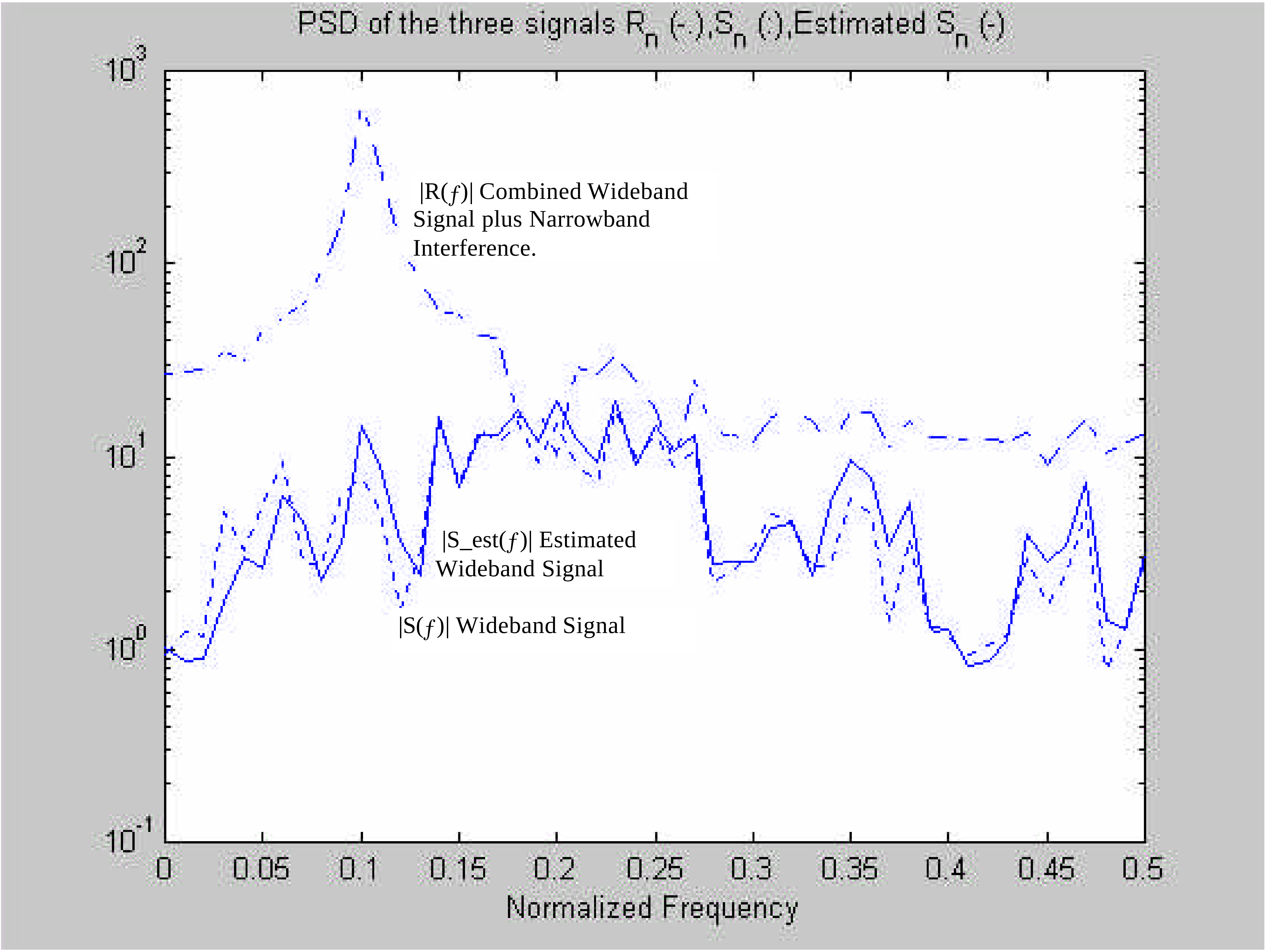
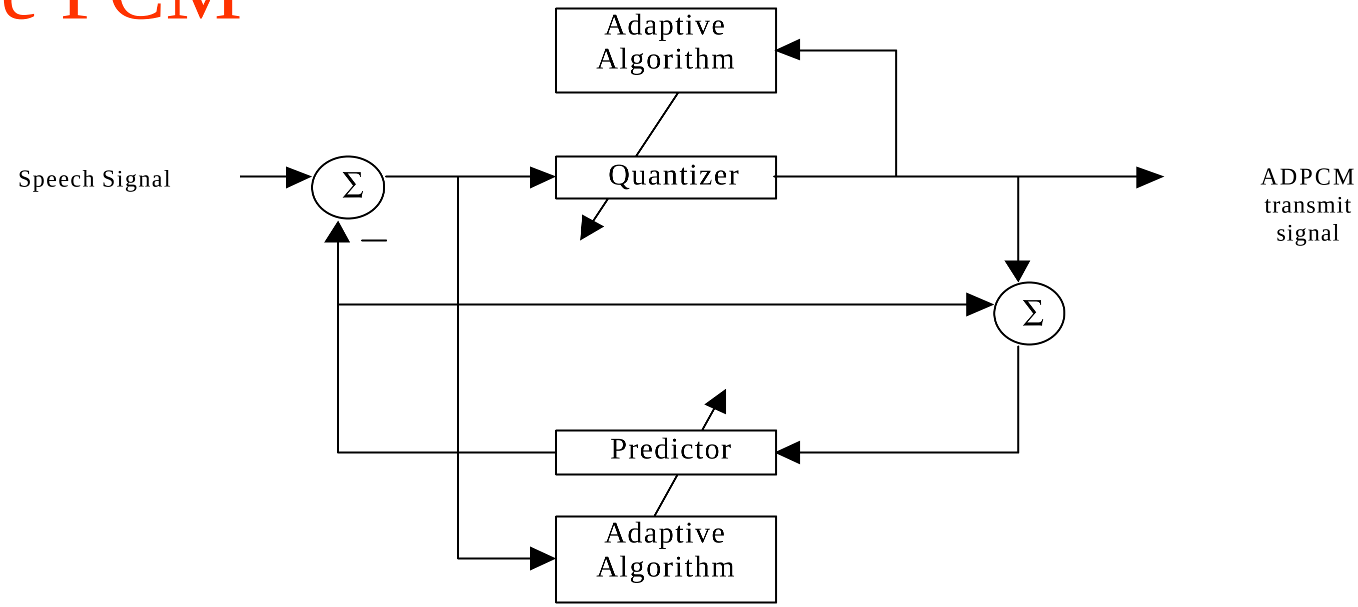




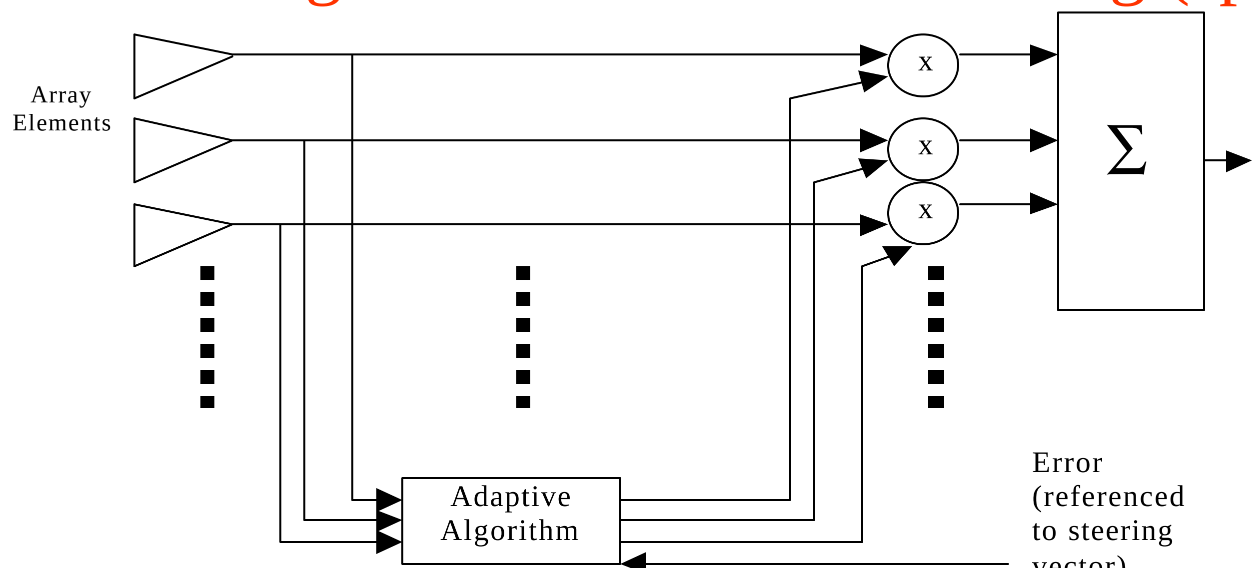

**Figure 8 Adaptive Filter for suppressing narrowband interference in a wideband signal**


PSD of the three signals Rn (-.),Sn (:),Estimated Sn (-)
|R(f)| Combined Wideband Signal plus Narrowband Interference.
|S_est(f)| Estimated Wideband Signal
|S(f)| Wideband Signal
Normalized Frequency
0
0.05
0.1
0.15
0.2
0.25
0.3
0.35
0.4
0.45
0.5
10^-1
10^0
10^1
10^2
10^3

# Adaptive PCM




# Adaptive Beam Forming and interference canceling (space)



# Meet some Families of Speech Coders

- Objective: to significantly reduce the number of bits which must be transmitted, but without creating objectionable levels of distortion
- Concerned with voice signal already band-limited to 4 kHz. max. and sampled at 8 kHz.
- The objective is *toll-quality* voice reproduction
- A few different strategies and algorithms used in voice compression:

| | |
|---|---|
| **Waveform Coders** | **PCM (pulse-code modulation), APCM (adaptive PCM)**<br>**DPCM (differential PCM), ADPCM (adaptive DPCM)**<br>**DM (delta modulation), ADM (adaptive DM)**<br>**CVSD (continuously variable-slope DM)**<br>**APC (adaptive predictive coding)**<br>**RELP (residual-excited linear prediction)**<br>**SBC (subband coding)**<br>**ATC (adaptive transform coding)** |
| **Hybrid Coders** | **MPLP (multipulse-excited linear prediction)**<br>**RPE (regular pulse-excited linear prediction)**<br>**VSELP (vector-sum excited linear prediction)**<br>**CELP (code-excited linear prediction)** |
| **Vocoders** | **Channel, Formant, Phase, Cepstral, or Homomorphic**<br>**LPC (linear predictive coding)**<br>**STC (sinusoidal transform coding)**<br>**MBE (multiband excitation), IMBE (improved MBE)** |

# Speech Coders Used Mobile Technologies:

- Vocoders are usually described by their output rate (8 kilobits/sec, etc.) and the type of algorithm they use. Here's a list of the vocoders used in currently popular wireless technologies:

| bits/sec | Algorithm | Standard (Year) | MOS |
|---|---|---|---|
| 64k | log PCM | CCITT G.711 (1972) | 4.3 |
| 32k | ADPCM | CCITT G.721 (1984) | 4.1 |
| 32k | LD-CELP | CCITT G.728 (1992) | 4.0 |
| 16k | APC | Inmarsat-B (1985) | n/avail |
| 13/7/4/2 v | QCELP | CTIA, IS-54/J-Std008 (1995) | n/avail |
| 13k | RPE-LTP | Pan-European DMR, GSM (1991) | 3.5 |
| 9.6k | MPLP | BTI Skyphone (1990) | 3.4 |
| 8k | EFRC | IS-136 (1997) TDMA enhanced | n/avail |
| 8k | VSELP | CTIA IS-54 (1993) TDMA | 3.5 |
| 6.7k | VSELP | Japanese DMR (1993) | 3.4 |
| 6.4k | IMBE | Inmarsat-M (1993) | 3.4 |
| 8/4/2/1 v | QCELP | Enhanced Vocoder, 1997 CDMA | n/avail |
| 8/4/2/1 v | QCELP | CTIA, IS-95 (1993) CDMA | 3.4 |
| 4.8k | CELP | US, FS-1016 (1991) | 3.2 |
| 2.4k | LPC-10 | US, FS-1015 (1977) | 2.3 |

# CONCLUSIONS AND LESSONS LEARNED

- Many applications of Adaptive Filters are applicable and necessary within the ATN and AMSS

-The dynamic nature of the system, mobility, makes adaptive systems more critical

-generic and general results are shown for a LMS and TLMS adaptive filters, with TLMS showing improvement