Implementation of LMS FIR based Adaptive Filter for System Identification

Tushant
Digital Signal Processing
SRH Hochschule Heidelberg
Heidelberg, Germany
11015370@STUD.HOCHSCHULE-HEIDELBERG.DE

Abstract—**An adaptive filter is a digital filter that can self-adjust its transfer function under the control of some optimizing algorithms. Least Mean Square (LMS) and Recursive Least Square (RLS) are the most used optimization algorithms. Although the RLS algorithm outperforms the LMS algorithm, it has a high computational complexity, making it unsuitable for most practical scenarios. As a result, the LMS algorithm, including its various variants, is the most feasible choice of adaptive filtering algorithm. As the underlying digital filter in the LMS algorithm, a transversal FIR filter is used. This paper is based on implementation of LMS algorithm based adaptive filter for the application of unknown system identification.**

Keywords-Adaptive Filter, Least Mean Algorithm (LMS), FIR Filter, Learning Curve, Convergence of filter coefficients, Estimation of h(unknown system Coefficient), System identification, MATLAB.

I. INTRODUCTION

In recent years, adaptive signal processing has attracted considerable attention and interest.

Time-varying digital signal processing systems, such as digital filters and similar structures with time-varying coefficients or "weights," are the most common type of adaptive signal processing system.

The term "adaptive" comes from people's desire to imitate living systems in nature, which adapt to their surroundings in a variety of spectacular ways and reflect the work of a master designer that we have only been able to duplicate in limited ways. Similar to the evolution of manned aircraft, the route from concept to reality in adaptive systems has brought us away from direct simulation. An adaptive signal filter generally mimics its natural counterpart as much or less like a bird, if anything.

The fundamentals of adaptive signal processing are adaptive control and iterative process mathematics where early attempts are undertaken to develop systems that adapt to your settings.

An adaptive filter that updates its transfer function according to an optimal method is a least mean squares (LMS) filter. An example of the intended output and the input signal are provided for the filter. The filter then determines the weights or coefficients of the filter, which provide the least average squares of an error between the output and the input signal.

II. ADAPTIVE FILTER

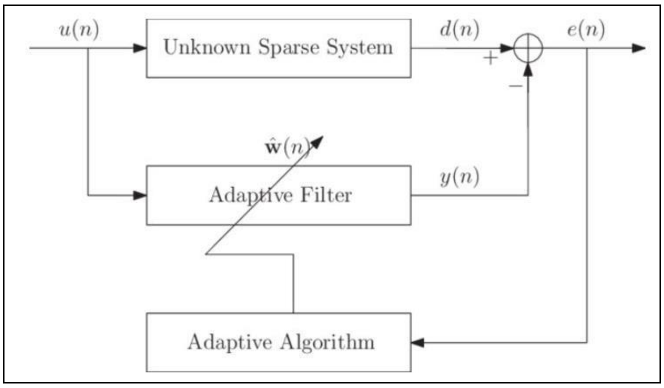
Signals of interest are typically polluted by noise or other signals that occupy the same frequency range. If the signal of interest and the noise lie in distinct frequency ranges, the desired signal can be removed by ordinary linear filters. However, when spectral overlays exist between signal and noise, or when statistics of the signal or interference with time vary, there are appropriate fixed coefficient filters.

A. ADAPTIVE ALGORITHM

Many methods for updating an adaptive filter weight are available. There is the Wiener filter, the optimum linear filter in terms of the mean squared error, and numerous methods, such as the steepest descent approach. There is also the least mean square algorithm, initially devised for use in artificial neural networks by Widrow and Hoff.

Finally, alternative techniques exist, such as the Kalman filter and the recursive least square method. The algorithm selection depends heavily upon the signals and the operating conditions as well as the needed time of convergence and available computing power.

GENERAL BLOCK DIAGRAM OF ADAPTIVE FILTER:



B. LEAST MEAN SQUARE ALGORITHM

The least-mean-square (LMS) algorithm, like the steepest-descent method, changes the weights by iteratively approaching the MSE minimum. In 1959 Windrow and Hoff

Identify applicable sponsor/s here. If no sponsors, delete this text box. (sponsors) created the LMS algorithm. LMS algorithm is known as a stochastic gradient algorithm. This means that it iterates each tap weight of transversal filter in the direction of the gradient of the squared magnitude of error with respect to the tap weight. The filter is only adapted based on the error at the current time. LMS algorithm is closely related to the concept of the stochastic approximation developed by Robbins and

Monro in 1951. The primary difference between the two is LMS algorithm uses the fixed convergence rate parameter to update the tap weight of the filter whereas stochastic approximation method uses the convergence parameter that in inversely proportional to the time n or power of n. There is another stochastic gradient algorithm which is closely related to LMS. This algorithm is called as Gradient Adaptive Lattice (GAL) developed by Griffiths in 1977. The difference between the LMS and GAL is only in underlying filtering structure, LMS uses transversal structure and GAL uses lattice structure. The LMS algorithm has an inherent limitation that it can search local minima only but not global minima. However, this limitation can be overcome by simultaneously initializing the search at multiple points. This algorithm is derived as follows.

$$C(n+1) = C(n) + \mu[p - Rw(n)] \quad (1)$$

The LMS algorithm is an approximate version of SDA which simply approximates R and p by replacing the expectation operator by instantaneous value. Thus,

$$R = E[x(n)x^T(n)] \approx x(n)x^T(n) \quad (2)$$

$$p = E[d(n)x(n)] \approx d(n)x(n) \quad (3)$$

Substituting (2) and (3) in Steepest Descent algorithm given in (1) gives,

$$C(n+1) = C(n) + \mu[d(n)x(n) - x(n)x^T(n)C(n)] \quad (4)$$

By separating out the common factor,

$$C(n+1) = C(n) + \mu\, x(n)\,[d(n) - x^T(n)C(n)] \quad (5)$$

Using the relationship

$y(n)=x^T(n)C(n)$ and $e(n)=d(n)-y(n)$, above equation can be written as

$$C(n+1) = C(n) + \mu\, x(n)\, e(n) \quad (6)$$

This equation represents popular LMS algorithm equation. This algorithm is of O(N) and requires 2N+1 multiplication and 2N addition per iteration. For stability of LMS algorithm convergence rate parameter μ satisfies the relationship,

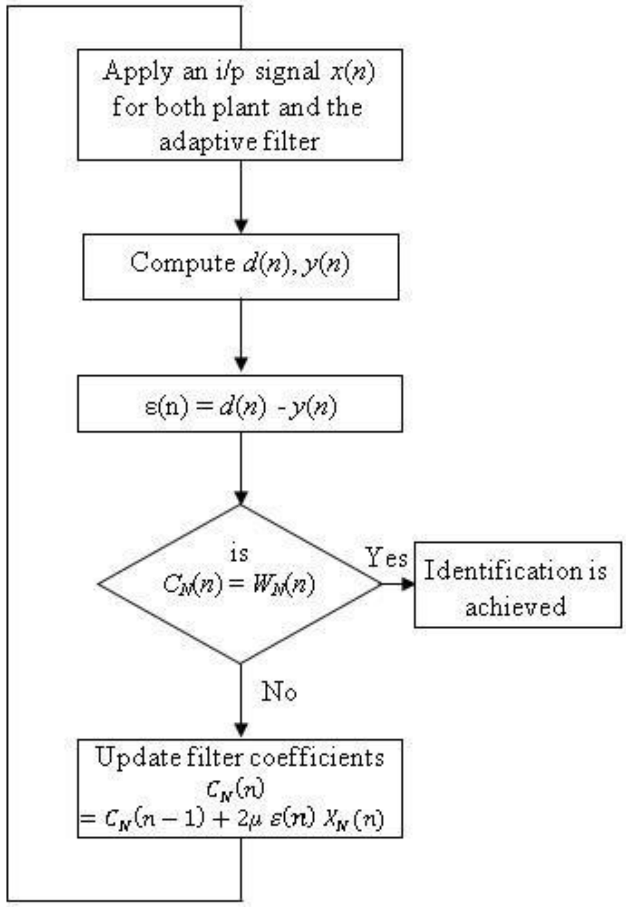
$$0 < \mu < \frac{2}{\lambda_{max}} \quad (7)$$

where λ_{max} is largest eigenvalues of the correlation matrix. In real time scenario, eigenvalues of the correlation matrix are not known which requires to modify (7) as,

$$0 < \mu < \frac{2}{\|x(n)\|^2} \quad (8)$$

where x (n) is called as Euclidean norm of the input vector which represents the power of the signal that is usually known or can be estimated a priority.

Flowchart of LMS algorithm.



Misadjustment

The misadjustment represents how far is the iterative solution at steady state condition from the optimal solution of cost function. Thus it is represented as,

$$M = \frac{J_{ex}}{J_{min}}$$

$$M = \frac{J_{ss} - J_{min}}{J_{min}} \quad (9)$$

where J_{ex} represents excess MSE, J_{ss} represents steady state MSE and J_{min} represents optimal MSE given by Weiner solution.

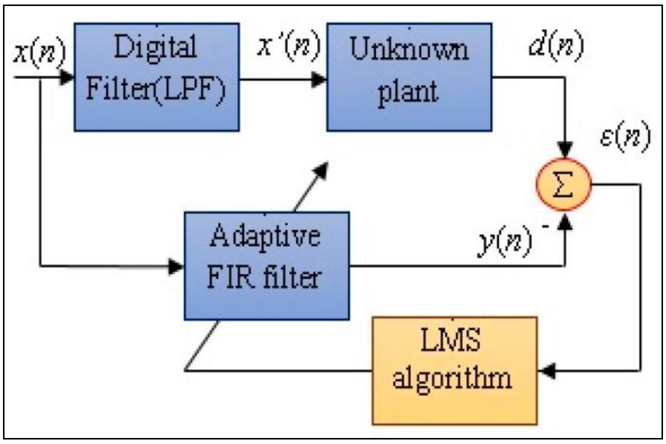
Robustness under H∞ criteria

Different researchers were working on the robust design issue. Zames in 1981 introduced H∞ or minimax criteria as robust index of performance. Hassibi in 1996 shown that LMS algorithm is optimal under H^∞ criteria which presented theoretical evidence for the robust performance of LMS algorithm.

C. SYSTEM IDENTIFICATION

A category of adaptive filtering, often commonly referred to as a mathematical modeling, finds an enormous array of applications, notably in the area of communication. The linear model that reflects the best fit for the unknown plant is supported with an adaptive filter. The system and the adaptive filter are connected in this configuration in parallel and are operated by the same input. The output of the plant is called the system's reaction. By removing the adaptive filter output from the desired output the error signal is obtained. When the system is dynamic, it is also time-variable or not stationary for adaptive filters.

Block Diagram of System identification using LMS



D. LEARNING CURVE

The Learning curve investigated the adaptation of the filter coefficient w as well as the behavior of the error over time. The error signal is not stationary during the transient (learning) phase. The mean square error is proportional to K, the number of iterations. Stationary mean Square Error of LMS.

$$J = J_{min} + J_{ex}$$

J_{ex}: is the excess error which is only in LMS is dependent on the step size μ

If the size of μ is not desirable then it causes variation in the mean square error.

III. IMPLEMENATION

To implement the tasks given on MATLAB.

- Task1: Implementation of LMS algorithm for N(number of coefficients)=1 and test for different step sizes and plot the graph for y[k]- filter output, d[k]- desired signal and x[k]- input signal. In below screenshot lms1 function code shared.

```
function y = lms1(x,d,N,mu)

iter = length(x);


w=zeros(N,iter);
k=(0:iter-1);
w_new=zeros(N,iter);
for i = 1:iter
    if i>1
    w(i) = w_new(i-1);
    end
   y(i) = w(i) * x(i);
   e(i) = d(i) - w(i) * x(i);
   w_new(i) = w(i) + mu * e(i) * x(i);


end
```

Task2: This task is dividing in three parts.

- plot the squared error e2 [k] (learning curve).
- Compare the learning curves with and without noise.
- Examine the convergence behavior of the filter coefficients plot w1=f(w2).
- In below screenshot lms2 function code shared.

```
function [y,e,w] = lms2(x,d,N,mu)
iter = length(x);

y = zeros(iter,1);
w = zeros(N,iter);
w_new = zeros(N,1);
e = zeros(iter,1);
k = (0:iter-1);
x_w = zeros(N,1);
MSE = zeros(1,1);
Cuml_E = zeros(1,1);

for i =  1:iter
    if i==1
        x_w(1)=  x(i);
    elseif  i > 1
        x_w(2) = x_w(1);
        x_w(1) = x(i);
    end

     y(i) = sum((w_new)'* x_w); % Adaptive filter output
    e(i) = d(i) - sum( (w_new)'* x_w) ;
    disp(e(i));
        w_new = w_new + mu * e(i) * x_w;
        w(1,i)= w_new(1);
        w(2,i)= w_new(2);
end
```

Task3: This task is divided into three subtasks.

- Loading the '.mat' file in the matlab code for feeding x and d values.
- Using system identification, estimate the length(L) of unknown system's coefficients.
- Using system identification, estimate the value of the unknown system's coefficients i.e. h.
- In below screenshot lms3 function code shared.

Note: N and mu(stepsize) can change, so the code should be generic.

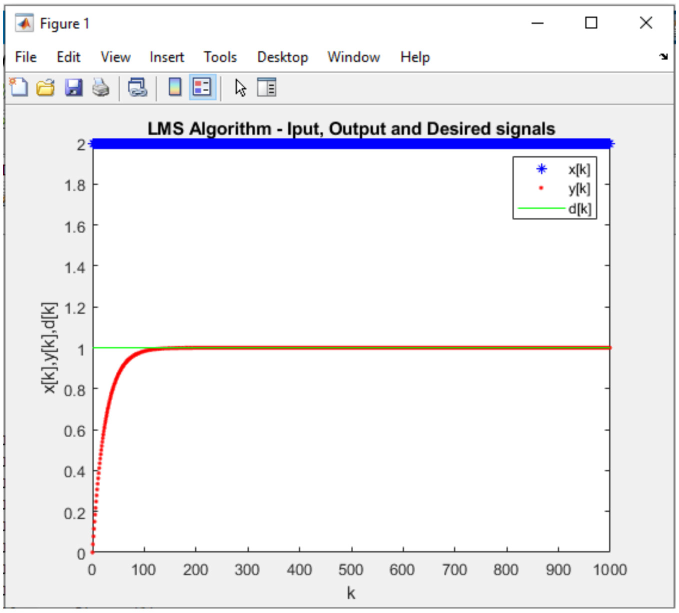
```
function [y,e,w] = lms3(x,d,N,mu)
iter = length(x);
y = zeros(iter,1);
w = zeros(N,iter);
w_new = zeros(N,1);
e = zeros(iter,1);
E = zeros(iter,1);
k = (0:iter-1);
x_w = zeros(N,1);

for i =  1:iter
    if i==1
        x_w(1)=  x(i);
    elseif  i > 1
        %x_w(N) = x_w(N-1);
        x_w = circshift(x_w,1)
        x_w(1) = x(i);
        % end
    end

    y(i) = sum((w_new)'* x_w);
    e(i) = d(i) - sum( (w_new)'* x_w) ;
    disp(e(i));
    w_new = w_new + mu * e(i) * x_w;
    for p = 1:N
        w(p,i)= w_new(p);
    end
```

IV. RESULTS AND GRAPHS

A. *Task 1. LMS Algorithm*

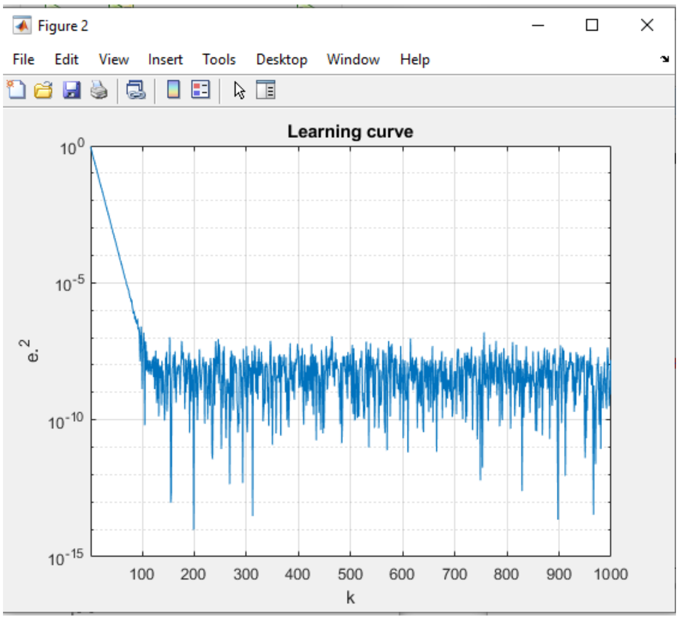


For x = 2,d=1, and k=1,2,3….999 and mu= 0.01

Here, we get output graph as the scalar product of X(n) and Transpose of Filter coefficient Wo.The graph consist of x(k) input signal,Desired signal d(k) and output signal y(k).

B. *Task 2. Learning Curve.*

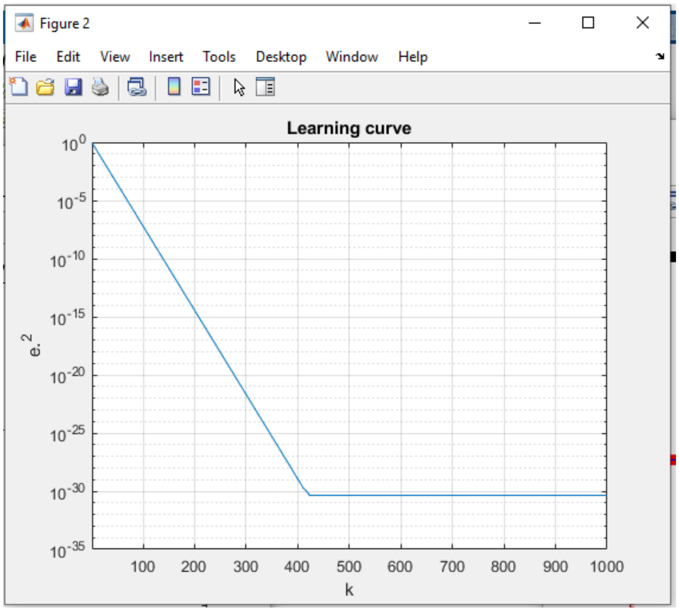
With Niose



For x = random signal with noise, d = d+ noise ,mu = 0.01 and k=1,2,3….999.

Here, we get output graph as the scalar product of X(n) with random noise added to see how it effects the output and Transpose of Filter coefficient Wo.The graph consist of x(k) input signal,Desired signal d(k) and output signal y(k).We can easily plot from the graph that there is a lot of noise in the output signal which might lead to loss of important signal data.

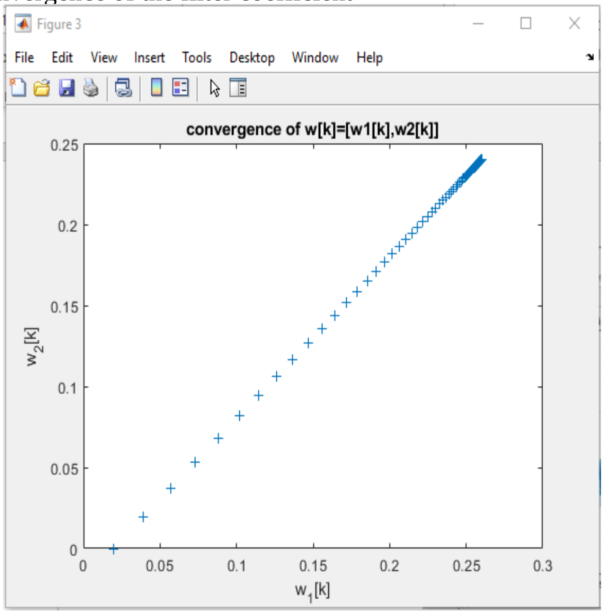
Without Noise



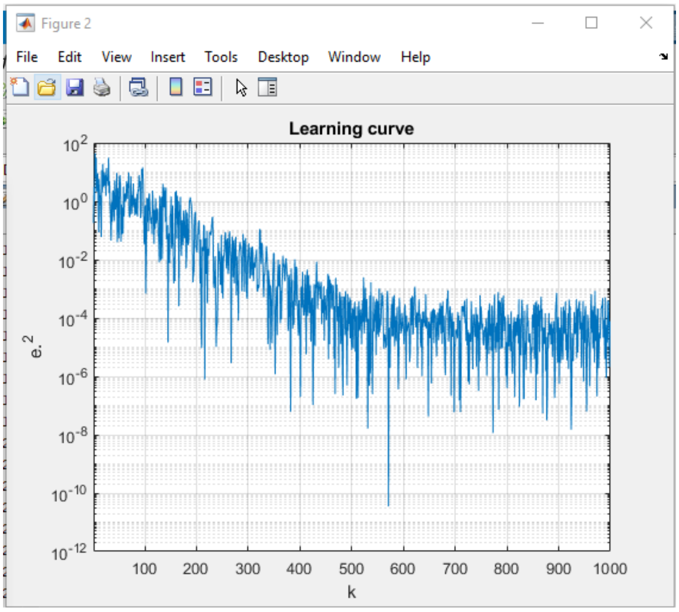
For x= random signal and d = 1, mu= 0.01 and k= 1,2,3,….999

Here, we get output graph as the scalar product of X(n) without random noise and Transpose of Filter coefficient Wo.The graph consist of x(k) input signal,Desired signal d(k) and output signal y(k).We can easily plot from the graph that there is no noise signal in the output signal of the learning curve.

Convergence of the filter coefficient



C. Task 3. System Identification



CONCLUSION

The project results show that the adaptive filter correctly estimates and converges on the unknown system coefficients. The effect of step size and number of iterations on filter performance, such as mean square error and estimate accuracy, has been widely analyzed. The reduction in step size reduces steady state error while also increasing the convergence time. Initially, the prediction process was carried out using the Least Mean Square (LMS) method, which identified the system based on its input and output responses. The prediction differed from the original signal by an acceptable amount. The following is the outcome of the analysis: The prediction procedure was initially carried out using the Least Mean Square (LMS) method, which identified the system based on its input and output responses. The prediction was within a reasonable range of the original signal. The analysis yielded the following results: The Least Mean Square (LMS) method was used to identify the system based on its input and output responses in the beginning of the prediction phase. The prediction matched the original signal within a tolerable range. The step size variation has a significant impact on the performance of the LMS algorithm. The step size determines the amount of correction to be applied to the input signal at each iteration in order to achieve the desired output. When considering a system with a sine wave as the desired signal, it is found that if the step size is small, the algorithm takes more iterations to reach the desired signal. As a result, we can deduce that if the step size is very small, the system convergence will necessitate a greater number of iterations. The system's convergence rate is faster if the step size is kept very large, but the system becomes unstable, which is not expected, thus the step size is chosen so that it is neither too little nor too large.

REFERENCES

[1] S.D.Shinde, Mrs.A.P.Patil, ,Mrs.P.P.Belagali3 ,"SYSTEM IDENTIFICATION USING ADAPTIVE FILTER ALGORITHMS," IOSR Journal of Electronics and Communication Engineering (IOSR-JECE) ISSN: 2278-2834-, ISBN: 2278-8735, PP: 54-59.

[2] B.Swapna Reddy, V. Rama Krishna on Implementation of Adaptive Filter Based on LMS Algorithm International Journal of Engineering Research & Technology (IJERT) ISSN: 2278-0181Vol. 2 Issue 10, October - 2013I.

[3] Ravi H Bailmare, S.J.Honale, Pravin V Kinge, "Design and Implementation of Adaptive FIR filter using Systolic Architecture" International Journal of Current Engineering and Technology, E-ISSN 2277 – 4106, P-ISSN 2347 - 5161

[4] Ali Akbar Siddiqui, S Hassan Ahmed,Habib Shaukat, M. Shafiq and Dr. M. Tahir Qadri on "Real Time Adaptive Filter Implementation For System Identification And Noise Elimination" Journal of Information & Communication Technology, Vol. 9, No. 2, (Fall 2015) 42-54

[5] Ibraheem Kasim Ibraheem, on "Adaptive System Identification Using LMS Algorithm Integrated with Evolutionary Computation" in Electrical Engineering Department, College of Engineering, Baghdad University

[6] Saurabh R. Prasad, Bhalchandra B. Godbole, on "Optimization of LMS Algorithm for System Identification" , DKTE Society's Textile and Engineering Institute, Ichalkaranji, India

[7] Ms.Kinjal Rasodia, Dr. Kiran Parmar on "Adaptive Filter Analysis for System Identification Using Various Adaptive Algorithms," in International Journal of Engineering Research & Technology(IJERT), ISSN: 2278-0181, Vol. 1. 3rd May 2012.

[8] Todd K. Moon, Wynn C. Stirling, "Mathematical Methods and Algorithms for Signal Processing .

[9] Simon Haykin, "Adaptive Filter Theory,Third Edition" J. Name Stand. Abbrev., in press..