System Identification through RLS Adaptive Filters

Hareeta Malani
Assistant Professor
ICT Department
M.L.V.T.E.C., Bhilwara.

ABSTRACT
System Identification is one of the most interesting applications for adaptive filters, especially for the Least Mean Square algorithm, due to its robustness and calculus simplicity. Based on the error signal, the filter's coefficients are updated and corrected, in order to adapt, so the output signal has the same values as the reference signal. The application enables remarkable developments and research, creating an opportunity for automation and prediction. In this paper we focus on parameters of system identification by changing design parameters such as forgetting factor, filter length, initial value of filter weight and input variance of filter through MATLAB/SIMULINK Software.

General Terms
System Identification, RLS Adaptive Filter.

Keywords
RLS Adaptive Filter, Forgetting Factor, Filter length, Filter weight, Input Variance.

1. INTRODUCTION
Adaptive filters are used for non stationary signals and environments or in applications where a sample-by-sample adaptation of a process or a low processing delay is required. The characteristics of digital filters can easily be changed by modifying the filter coefficients. This makes digital filters attractive in communications applications such as adaptive equalization, echo cancellation, noise reduction, speech analysis, and speech synthesis. The basic concept of an adaptive filter is shown in Figure 1.

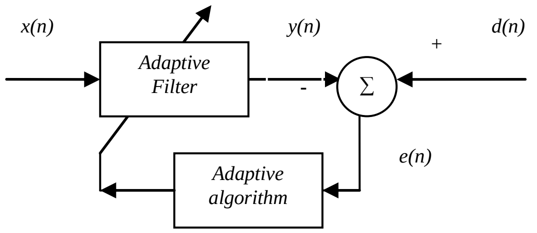

Figure 1: Adaptive filter

The objective is to filter the input signal, x(n), with an adaptive filter in such a manner that it matches the desired signal, d(n). The desired signal, d(n), is subtracted from the filtered signal, y(n), to generate an error signal. The error signal drives an adaptive algorithm that generates the filter coefficients in a manner that minimizes the error signal. The least-mean-square (LMS) or recursive-least-squares (RLS) adaptive filter are two of the most popular.

Identifying an unknown system has been a central issue in various application areas such as control, channel equalization, echo cancellation in communication networks and teleconferencing etc. Identification is the procedure of specifying the unknown model in terms of the available experimental evidence, that is, a set of measurements of the input output desired response signals and an appropriately error that is optimized with respect to unknown model parameters. Adaptive identification refers to a particular procedure where we learn more about the model as each new pair of measurements is received and we update the knowledge to incorporate the newly received information.

In the wide range of available adaptive algorithms, gradient descend methods, including the popular least mean squares (LMS) and recursive-least-squares (RLS) adaptive filter, are used.

2. RLS ADAPTIVE FILTER
The recursive least square error (RLS) filter is a sample-adaptive, time-update, version of the Wiener filter. For stationary signals, the RLS filter converges to the same optimal filter coefficients as the Wiener filter. For non-stationary signals, the RLS filter tracks the time variations of the process. The RLS filter has a relatively fast rate of convergence to the optimal filter coefficients. Figure 2 illustrates the configuration of an adaptive filter where $y(m)$, $x(m)$, $w(m) = [w_0(m), w_1(m), ..., w_{P-1}(m)]$ and P denote the filter input, the desired signal, the filter coefficient vector and the filter length respectively. The filter output can be expressed as

$$\hat{x}(m) = w^T(m)y(m) \qquad (1)$$

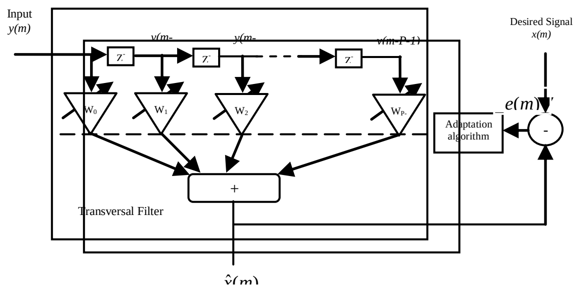

Figure 2: Configuration of an adaptive filter

Where, $\hat{x}(m)$ is an estimate of the desired signal $x(m)$. The filter error signal is defined as

$$e(m) = x(m) - \hat{x}(m)$$
$$= x(m) - w^T(m)y(m) \qquad (2)$$

The adaptation process is based on the minimization of the mean square error criterion defined as

$$E[e^2(m)] = E\{[x(m) - w^T(m)y(m)]^2\}$$
$$= r_{xx}(0) - 2w^T(m)r_{yx}(m) + w^T(m)R_{yy}(m)w(m) \qquad (3)$$

For stationary signals, the result of this minimization is given as

$$w = R_{yy}^{-1} r_{yx} \qquad (4)$$

Where, R_{yy} is the autocorrelation matrix of the input signal and r_{yx} is the cross-correlation vector of the input and the target signals. For a block of N sample vectors, the correlation matrix can be written as

$$R_{yy} = Y^T Y = \sum_{m=0}^{N-1} y(m)y^T(m) \qquad (5)$$

Where, $y(m)=[y(m), ..., y(m-P)]^T$. Now, the sum of vector product in Equation 5 can be expressed in recursive fashion as

$$R_{yy}(m)=R_{yy}(m-1)+y(m)y^T(m) \quad (6)$$

To introduce adaptability to the time variations of the signal statistics, the autocorrelation estimate in Equation 6 can be windowed by an exponentially decaying window:

$$R_{yy}(m)=\lambda R_{yy}(m-1)+y(m)y^T(m) \quad (7)$$

Where, λ is the so-called adaptation, or forgetting factor, and is in the range $0 < \lambda \leq 1$. Similarly, the cross-correlation vector is given by

$$r_{yx}=\sum_{m=0}^{N-1} y(m)x(m) \quad (8)$$

The sum of products in Equation 8 can be calculated in recursive form as

$$r_{yx}(m)=r_{yx}(m-1)+y(m)x(m) \quad (9)$$

Again this equation can be made adaptive using an exponentially decaying forgetting factor λ:

$$r_{yx}(m)=\lambda\, r_{yx}(m-1)+y(m)x(m) \quad (10)$$

For a recursive solution of the least square error Equation 10, we need to obtain a recursive time-update formula for the inverse matrix in the form

$$R_{yy}^{-1}(m)=R_{yy}^{-1}(m-1)+Update(m) \quad (11)$$

2.1 Recursive Time-update of Filter Coefficients

The least square error filter coefficients are

$$\begin{aligned} w(m)&=R_{yy}^{-1}(m)r_{yx}(m) \\ &=\Phi_{yy}(m)r_{yx}(m) \end{aligned} \quad (12)$$

Substituting the recursive form of the correlation vector in Equation 12 from Equation 10 yields

$$w(m)=\Phi_{yy}(m)[\lambda r_{yx}(m-1)+y(m)x(m)] \quad (13)$$

Now substitution of $k(m)=\Phi(m)y(m)$ and the recursive form of the matrix $\Phi_{yy}(m)=\lambda^{-1}\Phi_{yy}(m-1)-\lambda^{-1}k(m)y^T(m)\Phi_{yy}(m-1)$, in Equation 13 yields

$$w(m)=[\lambda^{-1}\Phi_{yy}(m-1)-\lambda^{-1}k(m)y^T(m)\Phi_{yy}(m-1)]\lambda r_{yx}(m-1)+k(m)x(m)] \quad (14)$$

Substitution of $w(m-1)=\Phi(m-1)r_{yx}(m-1)$ in Equation 14 yields

$$w(m)=w(m-1)-k(m)[x(m)-y^T(m)w(m-1)] \quad (15)$$

This equation can be rewritten in the following form

$$w(m)=w(m-1)-k(m)e(m) \quad (16)$$

Equation 16 is a recursive time-update implementation of the least square error Wiener filter.

2.2 The Steepest-Descent Method

The mean square error surface with respect to the coefficients of an FIR filter is a quadratic bowl-shaped curve, with a single global minimum that corresponds to the LSE filter coefficients. The steepest descent search is based on taking a number of successive downward steps in the direction of negative gradient of the error surface. The steepest-descent adaptation method can be expressed as

$$w(m+1)=w(m)+\mu\left[-\frac{\partial E[e^2(m)]}{\partial w(m)}\right] \quad (17)$$

where, μ is the adaptation step size. The gradient of the mean square error function is given by

$$\frac{\partial E[e^2(m)]}{\partial w(m)}=-2r_{yx}+2R_{yy}w(m) \quad (18)$$

Substituting Equation 18 in Equation 17 yields

$$w(m+1)=w(m)+\mu\left[r_{yx}-R_{yy}w(m)\right] \quad (19)$$

Let w_o denote the optimal LSE filter coefficient vector, we define a filter coefficients error vector $\tilde{w}(m)$ as

$$\tilde{w}(m)=w(m)-w_o \quad (20)$$

For a stationary process, the optimal LSE filter w_o is obtained from Wiener filter, as

$$w_o=R_{yy}^{-1}r_{yx} \quad (21)$$

From Equation 19 to 20, we get

$$\tilde{w}(m+1)=[I-\mu R_{yy}]\tilde{w}(m) \quad (22)$$

The parameter μ, the adaptation step size, controls the stability and the rate of convergence of the adaptive filter. Too large a value for μ causes instability; too small a value gives a low convergence rate. The correlation matrix can be expressed in terms of the matrices of eigenvectors and eigenvalues as

$$R_{yy}=Q\Lambda Q^T \quad (23)$$

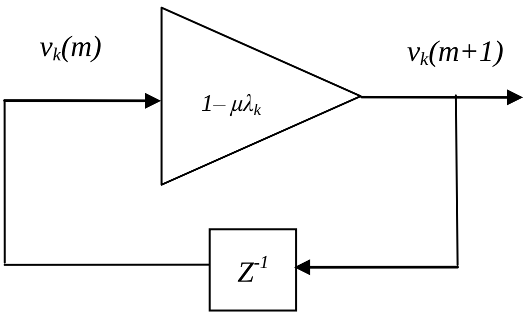

Figure 3: A feedback model of the variation of coefficient error with time

where, Q is an ortho-normal matrix of the eigenvectors of R_{yy}, and Λ is a diagonal matrix with its diagonal elements corresponding to the eigenvalues of R_{yy}. Substituting R_{yy} from Equation 23 in Equation 22 yields

$$\tilde{w}(m+1)=[I-\mu Q\Lambda Q^T]\tilde{w}(m) \quad (24)$$

Multiplying both sides of Equation 25 by Q^T and using the relation $Q^T Q = QQ^T = I$ yields

$$Q^T \tilde{w}(m+1) = [I - \mu\Lambda] Q^T \tilde{w}(m) \qquad (25)$$

Let, $v(m) = Q^T \tilde{w}(m)$ (26)

Then,

$$v(m+1) = [I - \mu\Lambda] v(m) \qquad (27)$$

As Λ and I are both diagonal matrices, Equation 27 can be expressed in terms of the equations for the individual elements of the error vector $v(m)$ as

$$v_k(m+1) = [I - \mu\lambda_k] v_k(m) \qquad (28)$$

Where λ_k is the k^{th} eigenvalue of the autocorrelation matrix of the filter input $y(m)$.

3. APPLICATIONS OF ADAPTIVE FILTERS

The most important driving forces behind the developments in adaptive filters throughout their history have been the wide range of applications in which such systems can be used. The major applications of adaptive filters are system identification includes adaptive noise canceling, inverse modeling, linear prediction and feed-forward control.

3.1 System Identification

In Figure 4, shows the general problem of *system identification*. In this diagram, the system enclosed by dashed lines is a "black box," meaning that the quantities inside are not observable from the outside. Inside this box is (1) an unknown system which represents a general input-output relationship and (2) the signal *η(n)*, called the *observation noise signal* because it corrupts the observations of the signal at the output of the unknown system.

Let d'(n) represent the output of the unknown system with x(n) as its input. Then, the desired response signal in this model is

d (n) = d'(n) + *η(n)* (29)

The task of the adaptive filter is to accurately represent the signal d'(n) at its output. If y(n) = d'(n), then the adaptive filter has accurately modeled or identified the portion of the unknown system that is driven by x(n). Let the unknown system and the adaptive filter both be FIR filters, such that

$d(n) = W^T_{opt}(n)X(n) + \eta(n)$ (30)

Where, $W_{opt}(n)$ is an optimum set of filter coefficients for the unknown system at time *n*. In the system identification there are two major applications, one is channel identification and another is adaptive noise cancellation.

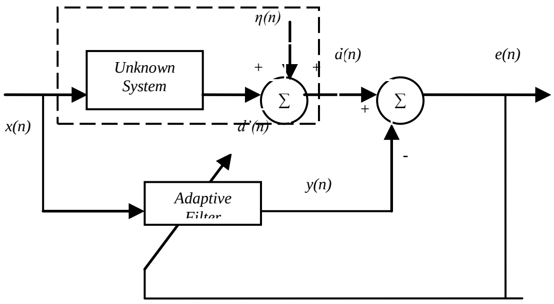

Figure 4: System Identification

3.2 Inverse Modelling

The *inverse modelling* system is shown in Figure 5. In this diagram, a *source signal* s(n) is fed into an unknown system that produces the input signal x(n) for the adaptive filter.

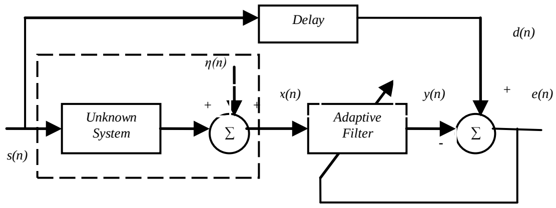

Figure 5: Inverse Modeling

The output of the adaptive filter is subtracted from a desired response signal that is a delayed version of the source signal, such that

d(n) = s(n-Δ) (31)

where, Δ is a positive integer value. The goal of the adaptive filter is to adjust its characteristics such that the output signal is an accurate representation of the delayed source signal.

3.3 Feedforward Control

Another problem area combines elements of both the inverse modeling and system identification tasks and typifies the types of problems encountered in the area of adaptive control known as feed forward control. Figure 6 shows the block diagram for this system, in which the output of the adaptive filter passes through a plant before it is subtracted from the desired response to form the error signal. The plant hampers the operation of the adaptive filter by changing the amplitude and phase characteristics of the adaptive filter's output signal as represented in e(n).

Thus, knowledge of the plant is generally required in order to adapt the parameters of the filter properly.

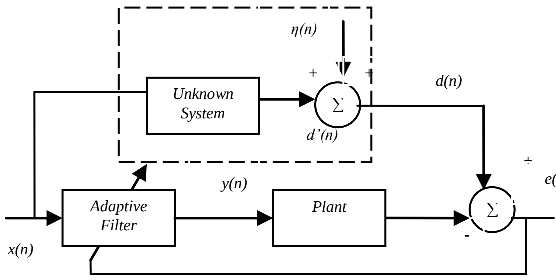

Figure 6: Feed forward Control

3.4 Linear Prediction

A third type of adaptive filtering task is shown in Figure 7. In this system, the input signal x(n) is derived from the desired response signal as

x(n) = d(n-Δ) (32)

Where, Δ is an integer value of delay. In effect, the input signal serves as the desired response signal, and for this reason it is always available. In such cases, the linear adaptive filter attempts to predict future values of the input signal using past samples, giving rise to the name *linear prediction* for this task.

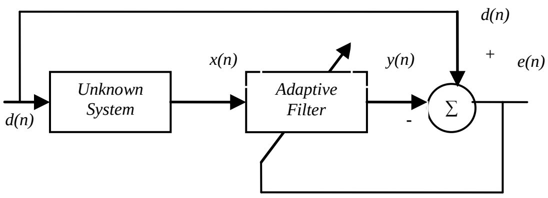

Figure 7: Linear Prediction

If an estimate of the signal x(n + Δ) at time *n* is desired, a copy of the adaptive filter whose input is the current sample x(n) can be employed to compute this quantity. However, linear prediction has a number of uses besides the obvious application of forecasting future events.

4. ANALYSIS OF RLS SYSTEM IDENTIFIER THROUGH MATLAB

The model of RLS adaptive for system identifier real time implementation through MATLAB is shown in Figure 8. The sine wave for input signal and noise as Gaussian noise are used. At input terminal of RLS filter put the noise signal and at desired signal put the summation of noise signal and input signal. Simulation of RLS adaptive linear predictor model gets the output signal i.e. converges to the input signal after the variation of forgetting factor, filter length, initial weight of filter and input variance of filter. Simulation also gives the prediction coefficient at the different values of the above parameters. Here, we present few waveforms from large collection.

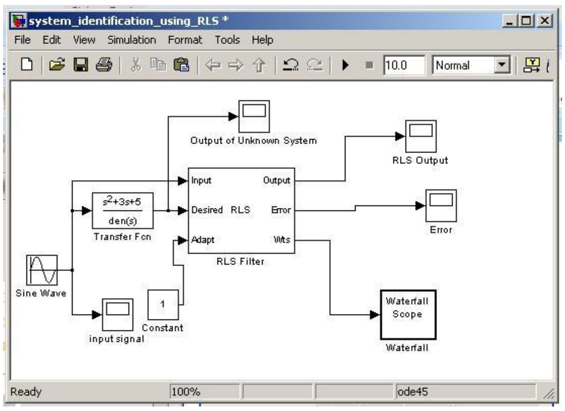

Figure 8: Model of RLS System Identifier

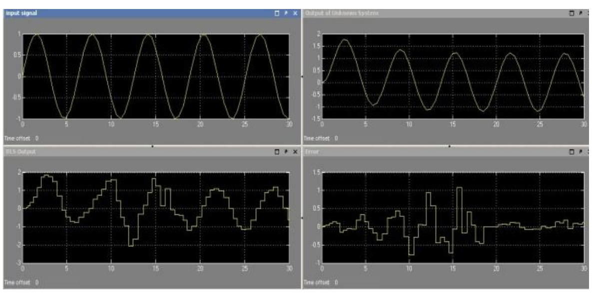

Figure 9: Waveforms of RLS System Identifier at forgetting factor = 0.3

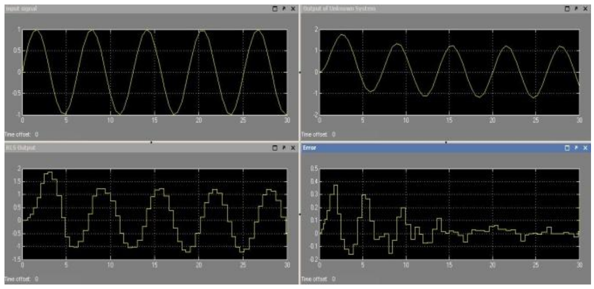

Figure 10: Waveforms of RLS Adaptive Linear Predictor at forgetting factor = 1

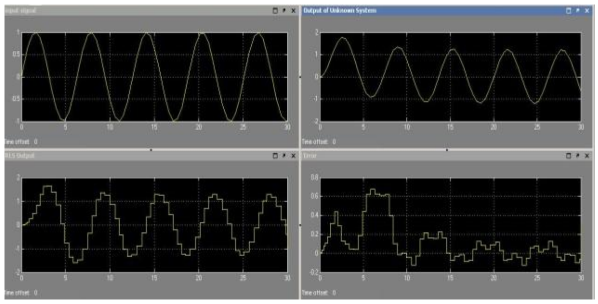

Figure 11: Waveforms of RLS System Identifier at filter length = 4 and initial value 0

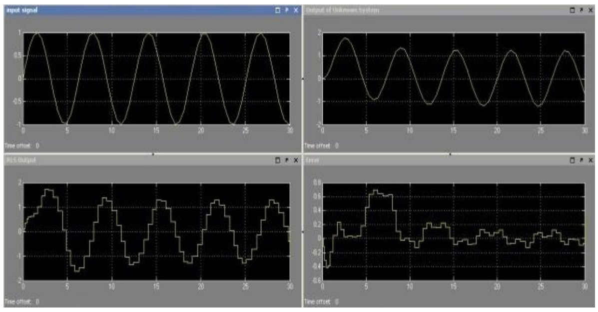

Figure 12: Waveforms of RLS System Identifier at filter length = 4 and initial value 1

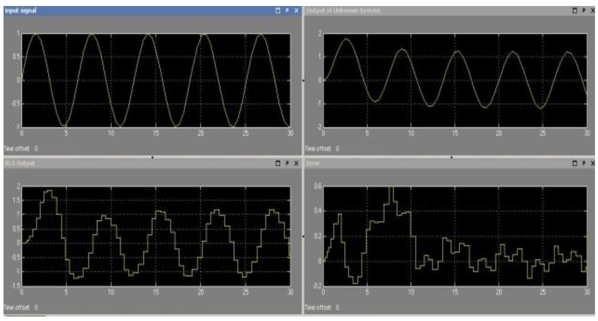

Figure 13: Waveforms of RLS System Identifier at filter length = 8 and initial value 0

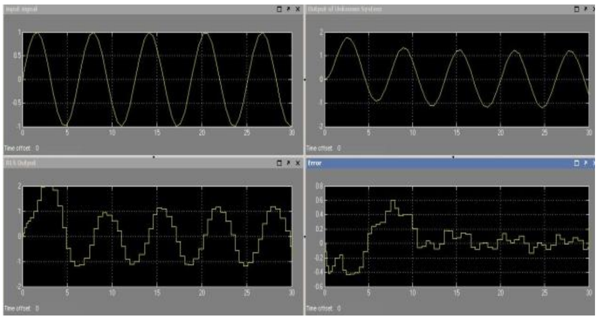

Figure 14: Waveforms of RLS System Identifier at filter length = 8 and initial value 1

The waveforms show the input signal, output of unknown system, RLS output signal and Error. The figures 9 and 10 show the waveforms of adaptive linear predictor at different forgetting factors 0.3 and 1.0 respectively. Therefore, after simulation and with the help of waveforms we can say if the forgetting factor increases then the output signal tends to the desired signal.

Similarly, Figures 11 to 14 shows the waveforms of adaptive linear predictor at different filter length 4, 8 for initial values 0 and 1 respectively. Hence we can say if the filter length increases then the output signal tends to the desired signal but the time it get to came up with desired signal increases and initial positive values produce more negative error.

The following table gives a few of the final filter coefficients in each case which are considerable.

4 coefficients	8 coefficients	16 coefficients	32 coefficients	50coefficients	60 coefficients	
0.0538	-0.4252	-0.1994	-0.1686	-0.1595	-0.1268	
0.5051	1.065	0.6931	0.6092	0.5874	0.5455	
-0.0599	0.5869	0.4637	0.4955	0.5061	0.5096	
0.9328	0.2307	0.3565	0.3956	0.4003	0.4064	
	0.2986	0.4337	0.3732	0.3636	0.3609	
	-0.0989	0.2005	0.1865	0.1913	0.2014	
	-0.5141	-0.2037	-0.074	-0.056	-0.0468	
	0.8652	-0.1938	-0.104	-0.1025	-0.1006	
		0.3053	0.1748	0.1474	0.1292	

5. CONCLUSION

Here, we have presented an overview of adaptive filters, emphasizing the applications and basic algorithms that have established and their real time implementation. In spite of the many contributions in the field, research efforts in adaptive filters continue at a strong pace. A model based design of system identification using RLS adaptive filter is implemented on MATLAB. The simulation model gives the variation in the output signal on different forgetting factors, filter lengths, initial value of filter weight and input variance of filter. After simulation, we conclude that if the filter length decreases or forgetting factor increases or initial value of filter weight decreases or input variance of filter decreases then the filter tends to its ideal state and get the output signal equivalent to the input signal; this is the phenomenon of linear prediction.

6. REFERENCES

[1] Vinay K. Ingle, John G. Proakis, Digital Signal Processing Using *MATLAB*, 2011.

[2] Jacob Benesty, Jingdong Chen, Yiteng Huang, Noise Reduction in Speech Processing, Springer, 2009.

[3] Ali H. Sayed, Adaptive Filters, John Wiley & Sons Limited, 2011.

[4] Kong-Aik Lee, Woon-Seng Gan, Sen M. Kuo, Subband *Adaptive Filtering*: Theory and Implementation, John Wiely & Sons Limited, 2009.

[5] Saxena, Gaurav; Ganesan, Subramaniam; Das, Manohar; "Real Time Implementation of Adaptive Noise Cancellation"; The proceedings of IEEE, 2008, pp. 431-436.

[6] Paulo Sergio Ramirez Diniz, *Adaptive filtering*: algorithms and practical implementation, Springer, 2008.