A New Prediction Algorithm: Flexible Switch for Combining LMS and RBF Adaptive Filters (CLMF)

M.REZA. POOSHIDEH¹, JAVAD. HADDADNIA² TAHERE .ROYANI³

¹,²,³Department of electrical engineering
Tarbiat Moallem University of Sabzevar
Sabzevar
Iran

m.poshideh@gmail.com; Haddania@sttu.ac.ir; tahere.royani@gmail.com

Abstract: Estimation and prediction are important tasks in the communication system. The LMS and RBF are suitable algorithms for prediction of linear and nonlinear model respectively. The LMS (Least Mean Square) algorithm is a linear adaptive filter, which has properties of slow convergence and good tracking in low SNR and RBF (Radial Basis Function) adaptive filter is a nonlinear filter that predict nonlinear behavior of signals. The LMS algorithm can predict each part of signal which has linear form better than the RBF adaptive filter. However, if linear combining of buffered signal can predict subsequent sample, the RBF adaptive filter have not results as well as the LMS algorithm. Suitable soft or flexible switch is needed for this purpose. In this paper, a stochastic gradient based switch is proposed based on error of the LMS and RBF algorithms. The proposed algorithm is configured for prediction of sinusoidal signal and chirp tracking problem. Experimental results show better performance compared to both the RBF and LMS algorithms in prediction problem and noisy chirp tracking.


1 Introduction

Adaptive filtering is utilized in numerous applications to help modeling of time variations of system parameters. In lack of a priori knowledge of the statistical model of the input signal, a wide range of algorithms has been developed. Among these, the least mean square, LMS [1], algorithm is very attractive, as it provides an efficient, robust and low-complexity solution. Also the simplicity of the LMS algorithm has made it an important benchmark for other algorithms. The ability of the LMS algorithm to operate in a nonstationary environment has been investigated by many authors [2]-[4]. However, in some parts of signals, a linear adaptive filter cannot work as well. This idea was presented as a preliminary version in [5].

In the past years many researchers has been work on combining adaptive filters. But all of these authors focused on combining analogous type of adaptive filters such as MLMS, CLMS [5]. We proposed a novel approach for combining linear and nonlinear adaptive filters. This method is sensible and useful, because it enables us to use advantages of heterogeneous adaptive filters such as LMS and RBF prediction algorithms. It should be noted that linear filters are useful to predict linear signals due to their linear nature, and this idea could be generalized for non-linear filters. That is they expected to have efficiency in prediction of non-linear signals. The LMS and RBF algorithms are explained briefly in section 2. Section 3 is devoted to the proposed algorithm. In section 4 simulation results are presented and conclusions are described in the final section.

2 The LMS and RBF algorithm

In this section, we briefly review the LMS and RBF algorithms.

2.1 The LMS algorithm

The LMS algorithm is an important member of the family of gradient algorithms. A significant feature of the LMS algorithm is its simplicity and good tracking properties in identification problem at low SNR. The LMS is a linear adaptive filtering algorithm that consists of a filtering process and an adaptation process according to the following equations:

\[ y_k = X_k^T w_k. \]  

(1)

Filtering process:

\[ W_{k+1} = W_k + \mu (d_k X_k - X_k^T W_k) \]

Adaptation process:

\[ e_k = d_k - y_k \]

(2)

The weight vector of the estimator at time index \( k \) is \( W_k = [w_k(1), w_k(2), ..., w_k(N)]^T \) and \( X_k \) is the N-element vector of the samples of a data sequence which is a stationary random process and N is the number of filter taps and \( e_k \) is the estimation error and \( d_k \) the desired response and \( \mu \) is the step size.
2.2. The RBF algorithm

The Radial basis function networks originated from multidimensional interpolation and feed forward models. The term ‘feed forward’ means that the neurons are organized in the form of layers in a layered neural network [9]. An RBF neural network has three layers including input layer, hidden layer and output layer. The input layer is composed of input data. The hidden layer transforms the data from the input space to the hidden space using a nonlinear function. The final layer (output layer), which is linear, yields the response of the network. The argument of the activation function of each hidden unit in an RBF network computes the Euclidean distance between the input vector and the center of that unit.

In the structure of RBF the input data $X$ is an $I$-dimensional vector, which is transmitted to each hidden unit. The activation function of hidden units is symmetric in the input space, and the output of each hidden unit depends only on the radial distance between the input vector $X$ and the center for the hidden unit. The output of each hidden unit $h_j$, $j=1, 2 \ldots J$, is given by

$$h_j(x) = \phi_j(||x - c_j||).$$ (3)

Where $||\cdot||$ is the Euclidean Norm, $c_j$ is the center of the neuron in the hidden layer, and $\phi()$ is the activation function, which is a nonlinear function and has many types, for example, Gaussian, multi quadric, thin-spline and exponential functions. We use the popular Gaussian function as the transform function in the hidden layer. The Gaussian activation function can be written as:

$$\phi_j(x) = \exp \left[ -\frac{||x - c_j||^2}{2\rho^2} \right].$$ (4)

Where $x$ is the training data, and $\rho$ is the width of the Gaussian function. A center and a width are associated with each hidden unit in the network. The weights connecting the hidden and output units are estimated using least mean square method. Finally, the response of each hidden unit is scaled by its connecting weights to the output units and then summed to produce the overall network output. Therefore, the $k^{th}$ output of the network $\hat{y}_k$ is:

$$\hat{y}_k = w_0 + \sum_{j=1}^{M} w_{jk} \phi_j(x).$$ (5)

Where $\phi_j(x)$ indicates the response of the $j^{th}$ hidden unit, $W_{jk}$ is the connecting weight between the $j^{th}$ hidden unit and the $k^{th}$ output unit, and $W_0$ is the bias term.

Appropriate centers and connection weights in the RBF network lead to a network that is capable of forming the best approximation to any continuous nonlinear mapping up to an arbitrary resolution. Such an approximation introduces best nonlinear approximation capability into the prediction model in order to accurately predict propagation loss over an arbitrary environment based on adaptive learning from measurement data.

The RBF network depicted in Fig.1 is a three-layer network whose output nodes from a linear combination of the activation (basis) functions computed by hidden layer nodes [6]-[8].

3 The proposed algorithm

We combine the LMS and RBF algorithms in a parallel form. The proposed algorithm is shown in Fig.2. The combination scheme consists of two LMS and RBF adaptive filter for signal prediction. The first adaptive filter is a linear algorithm (slow convergence and good tracking) that will provide a very good prediction of the signal in slowly changing situations. The second adaptive filter is a nonlinear algorithm (rapidly changing) which achieves fast convergence and good tracking properties in situations where rapid changes take place. Attention that, two component adaptive filters operate completely isolated and their coefficients are adapted to minimize their own quadratic errors using the standard LMS algorithm,

$$e_i(n) = d(n) - w_i^T(n) a(n).$$ (6)

d(n) is the desired output signal, and $u(n)$ are the inputs to the filter. Output and error of combined filter ($y(n)$ and $e(n)$) as convex combinations of the two adaptive filters as follows,

$$y(n) = \eta(n)y_1(n) + [1 - \eta(n)]y_2(n)$$
$$e(n) = d(n) - y(n)$$
$$= \eta(n)e_1(n) - [1 - \eta(n)]e_2(n).$$ (7)
Where \( y_1 \) and \( y_2 \) are outputs of the LMS and RBF adaptive filters respectively. Also \( e_1 \) and \( e_2 \) are output error that obtain from comparing between desired signal \( d(n) \) and outputs of filters. Regarding parameter \( \eta(n) \), we will define it via a sigmoid activation function [9].

\[
\eta(n) = \text{sgm}[a(n)] = \frac{1}{1 + e^{-a(n)}}.
\]

(8)

Where we update \( a(n) \) at each iteration to minimize the quadratic error of the combined filter by means of a minimum square error stochastic gradient algorithm [10] with step size \( \mu_a \).

\[
a(n+1) = a(n) - \frac{\mu_a}{2} \frac{\partial e^2(n)}{\partial a(n)} = a(n) - \beta \mu_a(n)
\times e(n) [e_1(n) - e_2(n)] \times \eta(n) [1 - \eta(n)].
\]

(9)

Using a sigmoid activation function for mixing parameter have two advantages: first, it constrain \( \eta(n) \) values between 0 and 1; second, its derivative, i.e., factor \( \eta(n) \), reduces the adaptation speed and the gradient noise near the endpoints 0 and 1, when the combined scheme is expected to perform, respectively, like the fast and slow filters without degradation.

In our implementation, we have limited \( a(n) \) values to the interval \([-4, 4]\), so the algorithm never stops because either \( \eta(n) \) or \( 1 - \eta(n) \) are then never too close to zero. When abrupt or fast changes appear, the RBF filter achieves a quadratic error lower than LMS filter, application of the learning rule (9) will increase \( a(n) \), and \( \eta(n) \) will approximate to one. So, in this situation, the combination algorithm acts as a RBF filter.

However, in slow changes intervals (part A in fig 3), the LMS filter will give better results in signal tracking relative to RBF adaptive filter, thus the control algorithm (9) decrease \( a(n) \) so that \( \eta(n) \) will approach to zero, and the combined filter will behave as the LMS filter.

For best operation \( \mu_a \) in (9) is obtained by follow equation:

\[
\mu_a(n) = [e_1(n)^{-1} - e_2(n)^{-1}].
\]

(10)

This equation means that growth coefficient for \( a(n) \) and \( \eta(n) \) dependent of difference between LMS and RBF error. In this algorithm we select \( \beta \) equal 500 to form of try and error.

4 Simulation results

In this section we will test the proposed algorithm (CLMF) over sinusoidal signal prediction and chirp tracking.

4.1 Application of CLMF algorithm in sinusoidal signal prediction

We use one cycle of sinusoid signal for description our method. Experimental results show us that linear filter can be good predict signal in linear situation (part A in figure illustrates error of two adaptive filters per one cycle of sin curve.

Nature of output error depicted in Fig. 3 shows us linear filter (LMS) track signal better than nonlinear filter (RBF) do it in curve form location of signal (part B in Fig. 3). This error configuration shows that maximum error in LMS filter occurred at position that RBF prediction error is minimized, and vice versa. This is major key and good idea for us to use combination both filter and obtain minimum error in final algorithm.

In the next step, we combine two LMS and RBF adaptive filter and illustrate this idea with experimental results that we achieved minimum MSE (mean square error) in signal prediction and good mixed two adaptive filter for access their advantage in signal prediction field.
Fig. 3. Comparison of LMS (dashed) and RBF (dotted) error in signal prediction. Part A shows location that the LMS prediction signal error is lower than the RBF error; part B shows location that the RBF prediction error is lower than the LMS error.

Fig. 4. The LMS error into sine prediction (step size of the LMS algorithm is $\mu = 0.262$).

Fig. 5. The RBF error into sine prediction (used RBF adaptive filter include 8 Gaussian PDF).

Fig. 6. The RBF error into sine prediction (used RBF adaptive filter include 8 Gaussian PDF).

The error configuration depicted in Fig. 6(a) shows that total MSE (mean square error) for combined adaptive filters decrease to 0.0017 while mean square error for LMS predicted signal equal 0.2727 and for RBF predicted signal is 0.0050.

Pay attention to this fact that MSE for LMS algorithm is very large (with step size $\mu = 0.262$), this matter shows our combined method is powerful and carefully even for one filter we have sometime instantaneous low error however the MSE is high, and this adaptive filter suitable for our purpose. Figure 6(b) shows that $\eta(n)$ coefficient how act smoothly and effectual for efficient combination. This factor followed from (8) when $a(n)$ obtained from (9). In figure 6(b) the position that $\eta(n)$ near one our combined adaptive filter act similar RBF signal prediction algorithms and for position $\eta(n)$ near zero combined filters similar LMS signal prediction algorithm.

4.2 Noisy chirp tracking using the CLMF algorithm

We used a dynamic mixture of RBF and LMS in a noise reduction application. Adaptive recovery of a chirp sinusoid buried in noise is a standard method because the chirp sinusoid represents a well-defined form of nonstationary. The estimation error for 6284 samples for 2 seconds with 1 kHz sample rate of chirp is shown in table I. In this table we illustrate the result of several different types of sinusoid chirp signals. The proposed approach is better than the RBF and LMS signal predictors. Table I compared between LMS and RBF signal prediction algorithm via our proposed methods for combination them. This table shows experimental result for six different signals that are mixed with noise.

The LMS algorithm that applied has constant step size $\mu$ equal 0.0264, and RBF signal prediction algorithm has eight Gaussian function as the transform function in the hidden layer that express with equation (4) and final output of this algorithm obtained from (5).

In the last section of this article we show the result of nosily sinusoid signal at signal prediction algorithm. With compare between there algorithm that discussed previous section.

Fig. 9 illustrate MSE (mean square error) for LMS, RBF and our proposed method for combination them. The signal selected for prediction is:

$$\sin(x) + 0.1 \times \text{rand}(1, \text{length}(x)) .$$

(11)

When length(x) divided into 6284 section.

5 Conclusion

In the present work, a new approach based on a dynamic mixture of the RBF and LMS algorithms was presented. Obtained results over sinusoidal chirp
tracking showed better performance relative LMS and RBF algorithms.

**Acknowledgment**

We would like to express our sincere thanks to Dr. Sadughi Yazdi for his valuable comments.

![Fig. 6. Performance of our combination method in signal prediction. 6(a) Total error of our combination adaptive filter with MSE equal 0.0017, and 6(b) state of η(n) coefficient for switching between LMS and RBF signal predictors.](image)

**Fig. 6.** Performance of our combination method in signal prediction. 6(a) Total error of our combination adaptive filter with MSE equal 0.0017, and 6(b) state of η(n) coefficient for switching between LMS and RBF signal predictors.

**TABLE 1**

Compare MSE for LMS predictor algorithm, RBF predictor algorithm and our proposed method in signal prediction for different types of sinusoid, noisily sinusoid and chirp signals.

<table>
<thead>
<tr>
<th>Type of Signals</th>
<th>LMS algorithm</th>
<th>RBF algorithm</th>
<th>The proposed method</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sin (x) [x=0:0.001:2*pi]</td>
<td>0.2727</td>
<td>0.0050</td>
<td>0.0025</td>
</tr>
<tr>
<td>sin(x)+.1*rand</td>
<td>8.88</td>
<td>8.62</td>
<td>7.43</td>
</tr>
<tr>
<td>Sinusoid chirp: fmin = 0 fmax = 30</td>
<td>11.85</td>
<td>91.73</td>
<td>3.87</td>
</tr>
<tr>
<td>Sinusoid chirp: fmin = 0 fmax = 60</td>
<td>11.93</td>
<td>531.17</td>
<td>3.95</td>
</tr>
<tr>
<td>Sinusoid chirp: fmin = 0 fmax = 150</td>
<td>12.26</td>
<td>1.74e+003</td>
<td>4.83</td>
</tr>
<tr>
<td>Sinusoid chirp: fmin = 0 fmax = 900</td>
<td>27.91</td>
<td>4.76e+003</td>
<td>20.56</td>
</tr>
</tbody>
</table>

![Fig. 7. Result of signal prediction with combined algorithm for chirp signal with maximum frequency of 100Hz.](image)

**Fig. 7.** Result of signal prediction with combined algorithm for chirp signal with maximum frequency of 100Hz.

![Fig. 8. Compare MSE for the algorithms in signal prediction for chirp signal.](image)

**Fig. 8.** Compare MSE for the algorithms in signal prediction for chirp signal.
Fig. 9. Compare MSE for their algorithm in signal prediction for noisily sinusoid signal.

References: