IMPLEMENTATION OF AFFINE PROJECTION SIGN ALGORITHM USING RECURSIVE TECHNIQUE

Mr.S.D.Shinde [1], Mrs.P.P.Belagali[2], Mrs.A.P.Patil [3]

[1]Dept. of Electronics & Telecomm Engg, JJMCOE Jaysingpur, 9561043781,

[2]Dept. of Electronics Engg, JJMCOE Jaysingpur. 9011077253.

[3] Dept. of Electronics Engg, JJMCOE Jaysingpur,9011977253.

ABSTRACT

Sign algorithms (SAs) are more robust against impulsive interference. However, most SAs suffer from slow convergence rate, especially for highly correlated input signals. In order to overcome this problem, recently, an affine projection SA (APSA) has been proposed, which exhibits fast convergence rate. In this letter, we first analyze APSA in detail and then apply a Recursive approach proposed for the affine projection algorithm (APA) to the APSA to reduce its computational complexity.

INDEX TERMS— Adaptive filtering, affine projection, efficient implementation, sign algorithm**.**

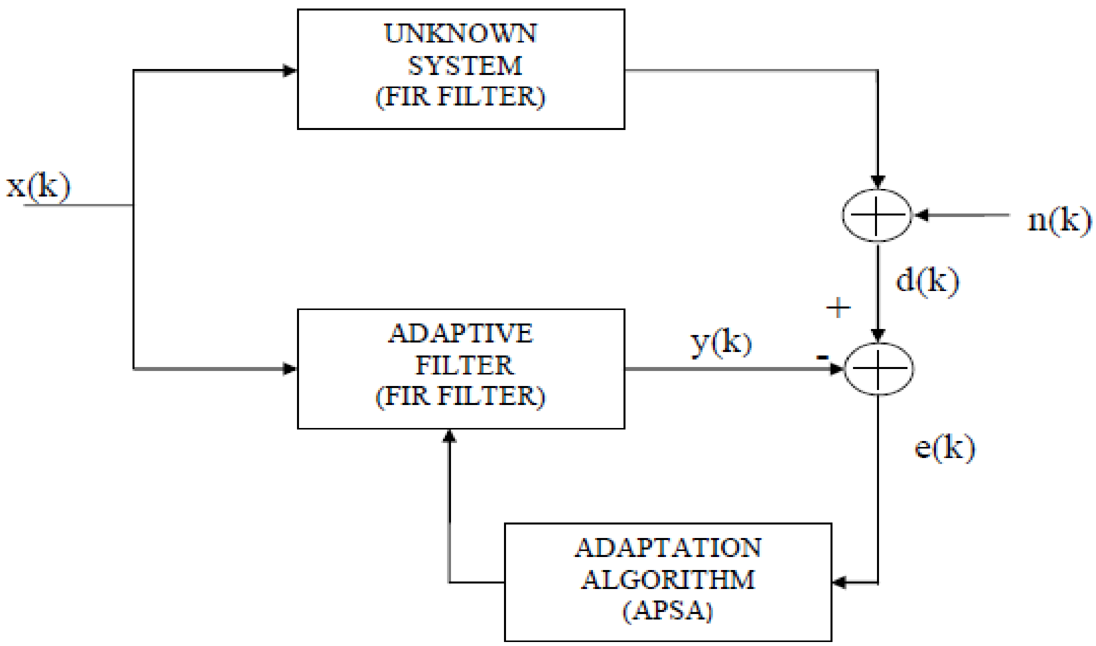
Corresponding Author: Mr.S.D.Shinde

INTRODUCTION

Adaptive filtering is a very important technique in a number of applications such as system identification, channel equalization, network and acoustic echo cancellation, and active noise control. An adaptive filter is characterized by its structure and algorithm. Once the structure of the adaptive filter has been selected, its convergence performance is fully dependent on its algorithm. It is well-known that adaptive filtering algorithms such as the least mean square (LMS) and normalized LMS (NLMS) have been widely used in the above applications. However, in some cases, the interference in the environment is impulsive, which degrades the performance of these adaptive filtering algorithms. Research has shown that classical sign algorithms (SAs) are robust against impulsive interference. However, they normally suffer from slow converge rate. Recently, a fast converging SA, called the affine projection SA (APSA), has been proposed, which combines the benefit of the affine projection algorithm (APA) and sign algorithm. Simulation results showed that the APSA can achieve improved performance on combating impulsive noise and speeding up convergence. Besides, the APSA is much simpler in implementation than the APA and does not have the numerical problems that exhibit in the classical fast affine projection (FAP) algorithm. Although the discussion of the computational complexity of the APSA was simply mentioned, it was not considered as the main issue.

In this letter, we first analyze the APA which is used for fast convergence rate as compared to LMS & NLMS algorithms but as the affine projection increases the computational complexity of the algorithm goes on increasing so APSA with the direct implementation in detail is studied and then apply the recursive approach which was proposed in for the APA is applied to the APSA to reduce its computational complexity. From the analysis, we will see that the computational complexity of the APSA with the efficient implementation can be reduced and is even lower than that of the classical fast affine projection (FAP) algorithm.

GENERALIZED BLOCK DIAGRAM



AFFINE PROJECTION ALGORITHM

APA is a generalization of NLMS. Where the coefficient update NLMS can be viewed as a rank-1 affine projection, a rank- N projection with N ³1 is made in APA. As the projection rank increases, the convergence speed of the adaptive filter increases as well, unfortunately so does the computational complexity. The Nth-order affine projection algorithm, in a relaxed and regularized form, can be defined as,

$$e(k) = s(k) - X^T(k)\ h(k-1)$$
$$\varepsilon(\mathrm{k}) = [\,X^T(\mathrm{k})\ \mathrm{X(k)}\ +\ \delta\mathrm{I}\,]^{-1}\ \mathrm{e(k)}$$
$$h(k) =\ h(k-1)\ +\ \mu_{APA}\ \ X^T(k)\ \varepsilon(k)$$

When the projection order of APA is 1, it is equivalent to NLMS. However, the convergence of APA gets better with the increased in the projection order and APA demonstrates very good convergence properties with colored excitation signals. The scalar d is the regularization parameter for the sample autocorrelation matrix inverse used in the calculation of the N-length normalized residual echo vector, $\varepsilon(k)$. Where $X^T(k)X(k)$ may have eigenvalues close to zero, creating problems for the inverse, $X^T(k)\ X(k) + \delta I$ has d as its smallest eigenvalue which, if large enough, yields a well behaved inverse. The step-size parameter, μ_{APA} is the relaxation factor. As in NLMS, the algorithm is stable for

$$0< \mu_{APA} < 2.$$

If N is set to one, relations (1), (2), and (3) reduce to the familiar NLMS algorithm. Thus, APA is a generalization of NLMS.

AFFINE PROJECTION SIGN ALGORITHM

Consider the desired signal y(k) that arises from the linear model. where $(.)^T$ denotes the transpose operation, w is the unknown system to be estimated,x(k) is the input signal vector of length L, and v(k) represents the background noise plus interference signal.

$$y(k) = x^T(k)w + v(k)$$

Grouping the M recent input signal vectors together forms the input signal matrix, i.e. X(k) Define the output vector and the *a priori* error vector, respectively, as

$$y'(k) = X^T(k)w'(k-1)$$

and

$$e(k)=y(k)-y'(k)$$

let $x_s(k) = X(k)sgn(e(k))$ where $sgn(.)$ represents the sign function. Then the update equation of the APSA can be written as

$$w'(k) = w'(k-1) + \mu \frac{x_s(k)}{\sqrt{x_s^T(k)x_s(k) + \varepsilon}}$$

where μ is the step-size parameter which controls the tradeoff between the convergence rate and final misalignment and is the regularization parameter which is used to overcome numerical difficulties. Table I describes the computational complexity. The algorithm requires one square root operation and one division at each time instant. Although sign operations are also required in the APSA, they are much simpler than multiplication and addition operations. It can be seen that the APSA with the direct implementation requires multiplications and additions at each time instant and that the computational complexity of the APSA increases with the increase of its projection order.

RECURSIVE AFFINE PROJECTION SIGN ALGORITHM

we see that the number of multiplications of the APSA with the direct implementation is mainly related to Step 1.

$$w'(k) = w'(k-1) + \mu \frac{X(k)sgn(e(k))}{\sqrt{sgn\left(e^T(k)\right)R(k)sgn(e(k)) + \varepsilon}}$$

Where $R(k) = X^T(k)X(k)$

$$y'(k) = X^T(k)w'(k-1)$$

$$y'(k) = X^T(k)w'(k-2) + \mu \frac{X^T(k)X(k-1)sgn(e(k-1))}{\sqrt{sgn\left(e^T(k-1)\right)R(k-1)sgn(e(k-1)) + \varepsilon}}$$

$$y'(k) = z(k) + \mu \frac{G(K)sgn(e(k-1))}{\sqrt{sgn\left(e^T(k-1)\right)R(k-1)sgn(e(k-1)) + \varepsilon}}$$

$z(k) = X^T(k)w'(k-2)$ and G(k)= $X^T(k)X(k-1)$

In this section, we employ an efficient implementation method to reduce the computational complexity of the APSA. In fast projection algorithm the recursive approach is used to calculate z(k),G(k),R(k).the same approach if applied to the APSA, an efficient implementation of the APSA can be obtained,

RESULTS OBTAINED

Here the input signal generated is of random form. As it is noticed that if the algorithm works properly for random input it obviously works for any type of input signal given to the system. The figure below on the left hand side shows the input signal given and on the right hand shows the APA algorithm output which is similar to that of the desired input signal

Here we have taken number of inputs as 10 and number of iterations as 100.The system has used APA adaptive algorithm. At every iteration it compares the desired output and output of the APA system and calculates the error signal. This error signal is used to generate the weight matrix which adds the correction on the input side so as to get desired signal at the output.

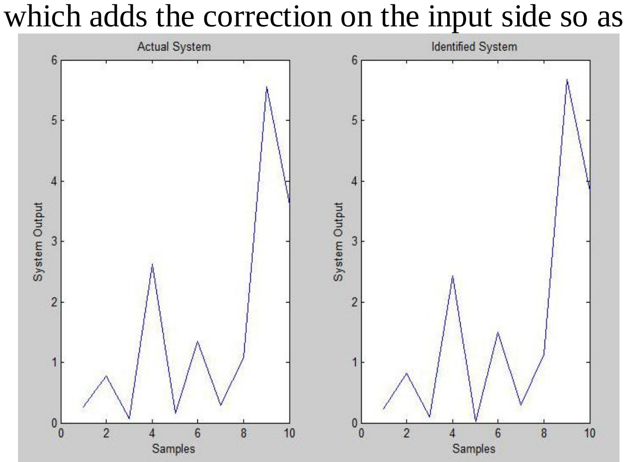


fig 2

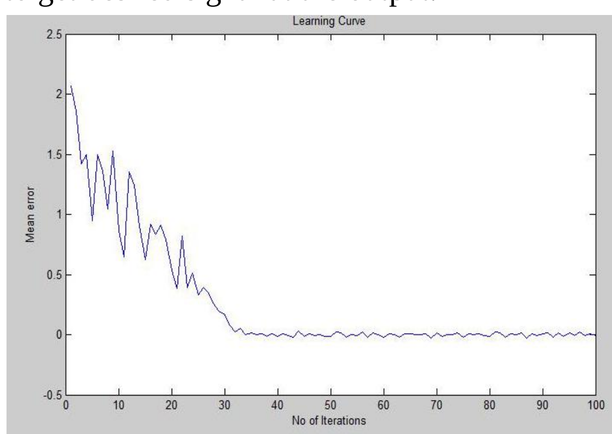


fig 2

The output is as shown in fig 1.If the graph of mean error versus number of iteration is calculated it is as shown above in fig2.here the order of projection is very important the no of iterations here is nothing the projection order of an apa.if the no of parameters are increased to 100 then the resultant output for apsa is as shown in fig 3 and fig 4

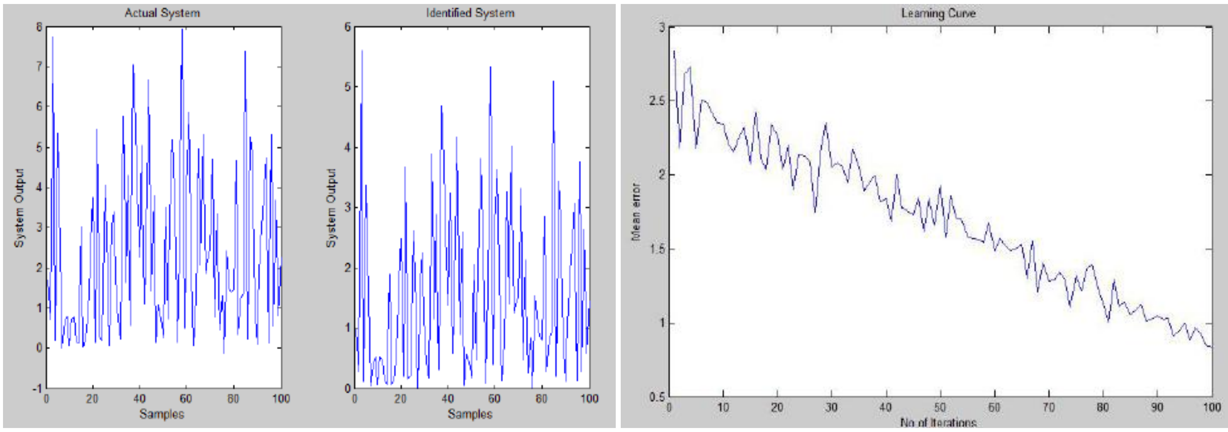


fig 4

As the time complexity of APA ghoes on increasing APSA is introduced when N=100 L=100 then APSA results are as shown in

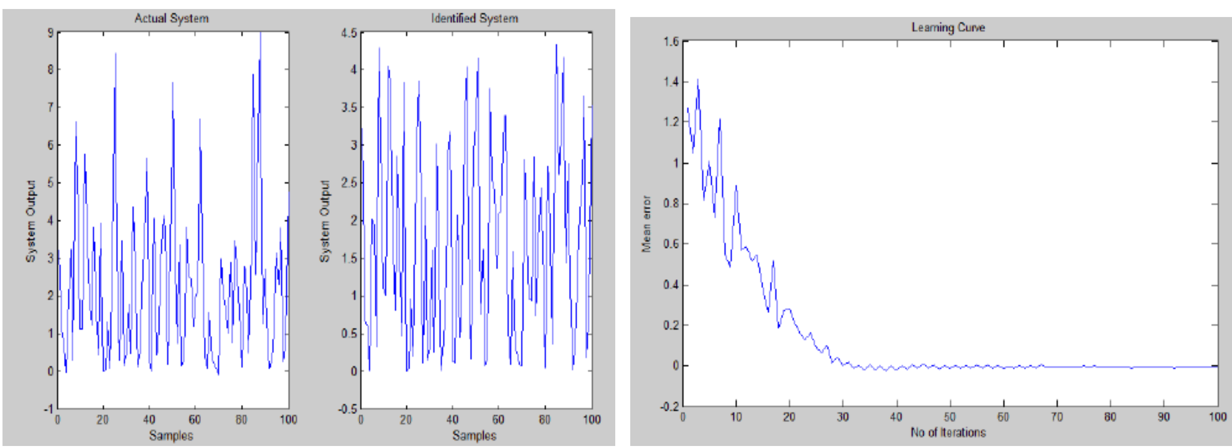


fig 4

Computational complexity of APSA is still more hence efficient APSA is introduced when the number of multiplications are reduced to much higher extent.the Shows the result when

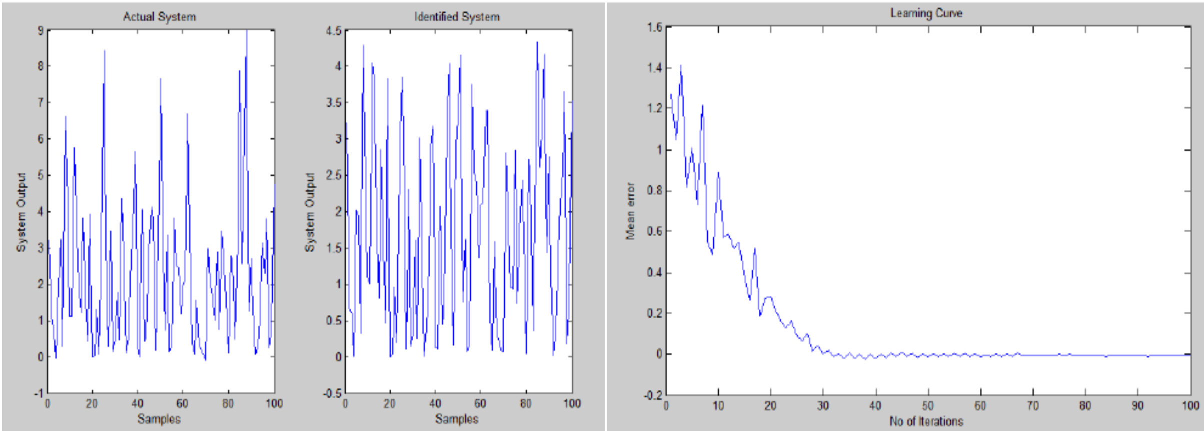


fig 4

CONCLUSION

It is seen that convergence rate of APA is fast as compared to NLMS but as projection order increases it degradeshence APSA algorithm is implemented. The computational complexity of the APSA with the direct implementation is relatively high and increases with increasing the projection order of the APSA. Then, a recursive approach which was first proposed for the APA was applied to the APSA to reduce its computational complexity. With this efficient implementation method, the APSA can be applied to adaptive filtering applications

REFERENCE

1) ***A learning method for system identification*** Nagumo, J.; Noda, A. Automatic Control, IEEE Transactions on *Volume: 12 , Issue: 3 Digital Object Identifier: 0.1109/TAC.1967.1098599 Publication Year: 1967 , Page(s): 282 - 287*

2) ***Efficient Implementation of the Affine Projection Sign Algorithm,*** Jingen Ni; Feng Li Signal Processing Letters, IEEE *Volume: 19 , Issue: 1 Digital Object Identifier: 0.1109/LSP.2011.2174784 Publication Year: 2012 , Page(s): 24 – 26*

3) ***Low-Complexity Implementation of the Affine Projection Algorithm*** Zakharov, Y.V.Signal Processing Letters, IEEE *Volume: 15 Digital Object Identifier: 10.1109/LSP.2008.2001111 Publication Year: 2008 , Page(s): 557 – 560*

4) ***A new sign algorithm for interference suppression in DS-CDMA systems*** Liping Sun; Guangrui Hu Communications Letters, IEEE *Volume: 7 , Issue: 5 Digital Object Identifier: 10.1109/LCOMM.2003.811194 Publication Year: 2003 , Page(s): 233 – 235*

5) ***A robust mixed-norm adaptive filter algorithm*** Chambers, J.; Avlonitis, A.Signal Processing Letters, IEEE *Volume: 4 , Issue: 2 Digital Object Identifier: 10.1109/97.554469 Publication Year: 1997 , Page(s): 46 – 48*

6) ***An Affine Projection Sign Algorithm Robust Against Impulsive Interferences*** Tiange Shao; Zheng, Y.R.; Benesty, J. Signal Processing Letters, IEEE *Volume: 17 , Issue: 4 Digital Object Identifier: 10.1109/LSP.2010.2040203 Publication Year: 2010 , Page(s): 327 – 330*

7) ***Proportionate Affine Projection Sign Algorithms for Network Echo Cancellation*** Zengli Yang; Zheng, Y.R.; Grant, S.L. Audio, Speech, and Language Processing, IEEE Transactions on *Volume: 19 , Issue: 8 Digital Object Identifier: 10.1109/TASL.2011.2125955 Publication Year: 2011 , Page(s): 2273 – 2284*

8) ***Low-complexity constrained affine-projection algorithms*** Werner, S.; Apolinario, J.A., Jr.; de Campos, M.L.R.; Diniz, P.S.R. Signal Processing, IEEE Transactions on *Volume: 53 , Issue: 12 Digital Object Identifier: 10.1109/TSP.2005.859348 Publication Year: 2005 , Page(s): 4545 – 4555*