International Journal of Scientific & Engineering Research, Volume 4, Issue 5, May-2013 821
ISSN 2229-5518

Noise Cancellation by Linear Adaptive Filter based on efficient RLS Lattice Algorithm

Kumar Shashi Kant, Amit Prakash

Abstract—This RLS Lattice algorithm is developed by using vector space treatment with the introduction of the Forgetting Factor such that it offers fast convergence and good error performance in the presence of noise. A linear Adaptive filter using this algorithm is developed which is having advantage that we can directly add the next stage to the existing filter without changing the previous filter coefficients to obtain the higher order filter. An audio signal with Gauss white noise is simulated in noise cancellation system on the MATLAB platform. This algorithm has very less computational complexity since it does not include any inverse matrix calculation and is highly efficient. The experimental results demonstrate the importance of using the newly updated iterations for the sensitivity of the noise estimation algorithm performance by using the forgetting factor in the algorithm.

Index Terms— Adaptive filters, Adaptive noise cancellation, Forgetting factor, Random variables, RLS Algorithm, White Gaussian Noise, Vector space.

1 Introduction

As compared to other adaptive algorithms Recursive Least Square (RLS) algorithm has rapid and exact convergence with a better noise handling capability across frequencies even when the Eigen value spread of the input signal correlation matrix is large. It extends the conventional scheme by adopting a numerical linear algebra random variable analysis without any mean operator [1]-[3]. A step further, RLS Lattice (RLSL) algorithm based adaptive filter is much more useful in audio processing and noise cancellation since the data processing at any instant of time for *(p+1)th* order requires only to add the new factor with the previous output signals of the *pth* order as the input to the next order. Further, the use of numerical linear algebra analysis gives better numerical performance of the algorithm and its stability such that the poles lies within the unity circle as explained in [4].

To handle the rapid change in the statistical properties, forgetting factor has been introduced in the algorithm. It gives rise to exponentially weighted growing window which further enhances the performance of the algorithm by handling the true error covariance matrix. In [5], it gives adaptive technique for noise reduction during non- speech segment but it performance is degraded during speech segment. The performance of the signal can be improved by using forgetting factor in the algorithm that leads to fast tracking. It exploits the concept of forgetting factor that may be required, due to model uncertainty, presence of unknown external disturbances, and time-varying nature of observed signal or non-stationary behaviour of observation noise [6].

In this paper, an RLS Lattice algorithm has been presented by using vector space treatment with the introduction of the forgetting factor such that it offers fast convergence and good error performance in the presence of noise. This algorithm is better than the other widely used adaptive algorithm, i.e., Least Mean Square (LMS) algorithm for noise cancellation in [7]. Finally, an improved hardware system for Adaptive Noise Canceller (ANC) is constructed and to achieve noise cancellation of audio signals.

2 Description of the Algorithm

2.1 RLS Algorithm

For the exponentially weighted least square method the tap-weight vector $\widehat{w}(n)$ at iteration $'n'$ is given by [1] as,

$$\widehat{w}(n) = \phi^{-1}(n) z(n) \qquad (1)$$

Where $\phi(n)$ is the correlation matrix of the input vector u(n) given by $\sum_{i=1}^{n} \lambda^{n-i} u(i) u^H(i)$, and $z(n)$ is the cross- correlation vector between input of the traversal filter and the desired response d(n) given by $\sum_{i=1}^{n} \lambda^{n-i} u(i) d^*(i)$.

Here the matrix $\phi(n)$ for tap weight update is given by Matrix Inverse Lemma [1], which states that

$$A^{-1} = B - BC(D + C^H BC)^{-1} C^H B$$

Where A and B are positive definite M-by-M matrices, C is M-by-N matrix and D is positive definite N-by-N matrix.

Since this algorithm uses inverse calculation of a matrix it has more computational complexity and difficult for hardware implementation.

2.2 RLS Lattice (RLSL) Algorithm

We consider, in general the prewindowed with exponentially weighted least square case, the input samples vector to the microphone be-

$$\bar{x}(n) = \begin{bmatrix} x(0) \\ x(1) \\ \vdots \\ x(n) \end{bmatrix}$$

ISSN 2229-5518

$$W_{k,p,n} = span\{z^{-k}\bar{x}(n), z^{-(k+1)}\bar{x}(n), \ldots\ldots, z^{-p}\bar{x}(n)\}$$

$W_{k,p,n}$ is the p-dimensional space spanned by the column input vectors up to n-th index where $z^{-k}\bar{x}(n)$ is given by

$$z^{-k}\bar{x}(n) = \begin{bmatrix} \left.\begin{matrix} 0 \\ 0 \\ \vdots \\ 0 \end{matrix}\right\} \text{k- zeroes} \\ x(0) \\ \vdots \\ x(n) \end{bmatrix}$$

$P_{k,p,n}$: Orthogonal projection operator with respect to space $W_{k,p,n}$ such that $P_{1,p,n}d(n) = \sum_{i=1}^{p} c_i \mathrm{z}^{-i}\,\bar{x}(n)$ where c_i is the optimal coefficient which linearly combines the vector on the space $W_{1,p,n}$.

Order update of forward error and backward error prediction:

$f_{p.n}$: *p-th* order forward prediction error operator on the space $W_{1,p,n}$ with *n-th* index as the current index

Now, $f_{p+1.n}$ is the forward error prediction operator on the space $W_{1,p+1,n}$, i.e.

$$f_{p+1.n} = \bar{x}(n) - P_{1,p+1,n}\bar{x}(n) \quad (2)$$

$W_{1,p+1,n}$ can be orthogonally decomposed into two spaces i.e., $W_{1,p,n}$ and the space spanned by the *(p+1)th* element vector $z^{-(p+1)}\bar{x}(n)$. The orthogonal projection error of the vector $z^{-(p+1)}\bar{x}(n)$ has the form $\begin{bmatrix} 0 \\ \ldots \\ b_{p,n-1} \end{bmatrix}$ denoted by $\breve{b}_{p,n-1}$.

$b_{p,n-1}$ is the *p-th* order backward error prediction operator on the space $W_{1,p,n-1}$ with *(n-1)-th* index as the current index.

Hence $f_{p+1.n}$ = sum of the forward error prediction operator on the space $W_{1,p,n}$ and the forward error prediction operator on the space spanned by $z^{-(p+1)}\bar{x}(n)$ which is given by

$$f_{p+1.n} = \bar{x}(n) - \left[P_{1,p,n}\bar{x}(n) + P_{\breve{b}_{p,n-1}}\bar{x}(n)\right]$$
$$= f_{p.n} - \frac{\langle \bar{x}(n), \breve{b}_{p,n-1}\rangle}{\left\|\breve{b}_{p,n-1}\right\|^2}\breve{b}_{p,n-1} \quad \textbf{(3)}$$

$\frac{\langle \bar{x}(n), \breve{b}_{p,n-1}\rangle}{\left\|\breve{b}_{p,n-1}\right\|^2}\breve{b}_{p,n-1}$ is the projection error of $\bar{x}(n)$ on $\breve{b}_{p,n-1}$.

$\langle \bar{x}(n), \breve{b}_{p,n-1}\rangle$ is the inner product of the vectors $\bar{x}(n)$ and $\breve{b}_{p,n-1}$ which gives the correlation between the two vectors denoted by $\Delta_{p,n}$. Thus

$$\Delta_{p,n} = \langle \bar{x}(n), \breve{b}_{p,n-1}\rangle \quad (4)$$

$$\left\|\breve{b}_{p,n-1}\right\|^2 = \sigma_{p,n-1}^{b^2} \quad (5)$$

$\sigma_{p,n-1}^{b^2}$ is the backward error variance. There may be chance that this variance becomes zero and therefore this should be initialized with a small positive value in the algorithm.

Now (3) transforms to

$$f_{p+1.n} = f_{p.n} - \frac{\Delta_{p,n}}{\sigma_{p,n-1}^{b^2}}\breve{b}_{p,n-1} \quad (6)$$

The present component of the vector $f_{p+1.n}$ comes out to be

$$f_{p+1}(n) = f_p(n) - \frac{\Delta_{p,n}}{\sigma_{p,n-1}^{b^2}} b_p(n-1) \quad (7)$$

Similarly, *(p+1)th* order backward error prediction is given by

$$b_{p+1.n} = \breve{b}_{p,n-1} - \frac{\Delta_{p,n}}{\sigma_{p,n}^{f^2}} f_{p.n} \quad (8)$$

$$\sigma_{p,n}^{f^2} = \left\|f_{p,n}\right\|^2 \quad (9)$$

$\sigma_{p,n}^{f^2}$ is the forward error variance and will never approach zero since it is the square of the error which is projected by the present component on the past p- components.

The present component of the vector $b_{p+1.n}$ comes out to be

$$b_{p+1}(n) = b_p(n-1) - \frac{\Delta_{p,n}}{\sigma_{p,n}^{f^2}} f_p(n) \quad (10)$$

With $p = 0$ in (6) and (8), and comparing the vectors $f_{0.n}$ and $b_{0.n}$ will be

$$f_{0.n} = b_{0.n} = \bar{x}(n) \quad (11)$$

The present component or the last component of the vectors $f_{p.n}$ and $b_{p.n}$ is $f_p(n)$ and $b_p(n)$ for $p = 0$ respectively which from (11) leads to

$$f_0(n) = b_0(n) = x(n) \quad (12)$$

For $n \geq p - 1$, the structure of RLSL filter follows from (7), (10) and (12) is given in Fig. 1.

The values of $\boldsymbol{K_{p,n}^f}$ and $\boldsymbol{K_{p,n}^b}$ in Fig.1 are both function of order and time that needs to be updated respectively.

$$\boldsymbol{K_{p,n}^f} = \frac{\Delta_{p,n}}{\sigma_{p,n}^{f^2}} \quad (13)$$

And, $\boldsymbol{K_{p,n}^b} = \frac{\Delta_{p,n}}{\sigma_{p,n-1}^{b^2}}$ (14)

ISSN 2229-5518

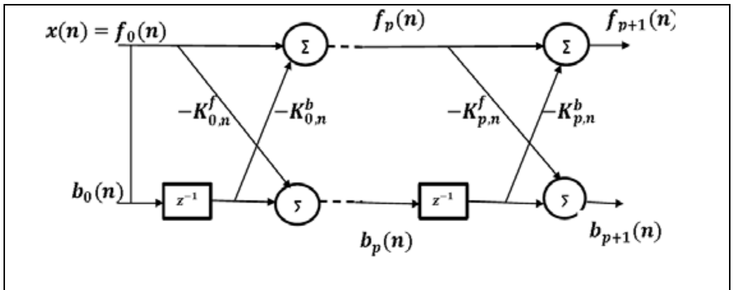

Fig.1 Schematic of RLSL Filter

From (13) and (14) forward error variance and backward error variance requires time and order update for the recursive value of $\boldsymbol{K^f_{p,n}}$ *and*$\boldsymbol{K^b_{p,n}}$ respectively.

***Time update of variance*:**

From (5), (9) and (11)

$$\sigma^{f^2}_{0,n} = \sigma^{b^2}_{0,n} = \|\bar{x}(n)\|^2 \qquad (15)$$

$\|\bar{x}(n)\|^2$ is the auto-correlation matrix of $\bar{x}(n)$ with exponential weighting factor or forgetting factor "λ" defined in pp.437 of [1].

$$\begin{aligned}\|\bar{x}(n)\|^2 &= [\lambda^n x^2(0) + \lambda^{n-1}x^2(1) + \cdots + \lambda x^2(n-1) + x^2(n)] \\ &= \lambda[\lambda^{n-1}x^2(0) + \lambda^{n-2}x^2(1) + \cdots + x^2(n-1)] + x^2(n)\end{aligned}$$

Hence,

$$\sigma^{f^2}_{0,n} = \sigma^{b^2}_{0,n} = \lambda\sigma^{f^2}_{0,n-1} + x^2(n) \qquad (16)$$

Order update of variance:

From (9),

$$\sigma^{f^2}_{p+1,n} = \left\|f_{p+1,n}\right\|^2$$

$\sigma^{f^2}_{p+1,n}$ is the inner product of $f_{p+1,n}$ and itself which gives the auto-correlation vector of $f_{p+1,n}$ which gives

$$\begin{aligned}\sigma^{f^2}_{p+1,n} &= \sigma^{f^2}_{p,n} - \frac{\Delta_{p,n}}{\sigma^{b^2}_{p,n-1}}(\Delta_{p,n}) \\ &= \sigma^{f^2}_{p,n} - \frac{\Delta^2_{p,n}}{\sigma^{b^2}_{p,n-1}} \qquad (17)\end{aligned}$$

Similarly, $\sigma^{b^2}_{p+1,n} = \sigma^{b^2}_{p,n-1} - \frac{\Delta^2_{p,n}}{\sigma^{f^2}_{p,n}}$ (18)

Time update for $\Delta_{p,n}$***:***

Consider the space $U_{\pi,n}$ such that

$$U_{p,\pi,n} = span\{z^{-1}\bar{x}(n), z^{-2}\bar{x}(n), \dots\dots, z^{-p}\bar{x}(n), \pi(n)\}$$

Where $\pi(n)$ is the pinning vector defined as $[0 \quad 0 \quad \dots \quad 1]^T$

And, $U_{p,\pi,n-1} = \left[\begin{bmatrix} z^{-1}\bar{x}(n-1) \\ \dots \\ 0 \end{bmatrix}, \quad \dots \begin{bmatrix} z^{-p}\bar{x}(n-1) \\ \dots \\ 0 \end{bmatrix}, \pi(n)\right]$

The *p-th* order backward error projection $b_{p,\pi,n}$ is given by

$$b_{\pi,n} = \begin{bmatrix} \tilde{b}_{p,n-2} \\ \dots \\ 0 \end{bmatrix} = \tilde{b}_{p,n-1} - \frac{b_{p,n-1}}{\vartheta_{p,n}}\rho \qquad (19)$$

$$where, \vartheta_{p,n} = error\ variance\ of\ \pi(n)$$

$$\rho = \ projection\ error\ of\ \pi(n)$$

From (19),

$$\tilde{b}_{p,n-1} = \begin{bmatrix} \tilde{b}_{p,n-2} \\ \dots \\ 0 \end{bmatrix} + \frac{b_{p,n-1}}{\vartheta_{p,n}}\rho$$

From (4) and (19),

$$\begin{aligned}\Delta_{p,n} &= \left\langle \bar{x}(n), \begin{bmatrix} \tilde{b}_{p,n-2} \\ \dots \\ 0 \end{bmatrix} + \frac{b_{p,n-1}}{\vartheta_{p,n}}\rho \right\rangle \\ &= \lambda\langle \bar{x}(n-1), \tilde{b}_{p,n-2} \rangle + \frac{f_p(n)b_p(n-1)}{\vartheta_{p,n}} \\ &= \lambda\Delta_{p,n-1} + \frac{f_p(n)b_p(n-1)}{\vartheta_{p,n}} \qquad (20)\end{aligned}$$

The present component or the last component ($\Delta_p(n)$) of vector $\Delta_{p,n}$ can be derived from (19) as

$$\Delta_p(n) = \lambda\Delta_p(n-1) + \frac{f_p(n)b_p(n-1)}{\vartheta_p(n)} \qquad (21)$$

The order update for $\vartheta_p(n)$ which is correlation of the error variance of $\pi(n)$ is given by

$$\vartheta_{p+1}(n) = \vartheta_p(n) + \frac{b_p^2(n-1)}{\sigma^{f^2}_{p,n}} \qquad (22)$$

The summary of the algorithm is depicted in Table.1.

3 Experimental Results

The primary input to the ANC is the desired signal x(n) mixed with the white Gaussian noise signal v(n). The mixed signal is passed through the adaptive filter based on RLS Lattice algorithm which recursively adjusts the filter coefficients to get the noise free output y(n) which matches with x(n) desired signal. The block diagram representation of the ANC is shown in Fig. 2.

The simulation is done on the MATLAB platform for the sinusoidal signal in one case and for the audio signal in the second case. In both the cases the order of the filter $p = 8$, forgetting factor $\lambda = 0.89$, $\delta = 0.01$.

ISSN 2229-5518

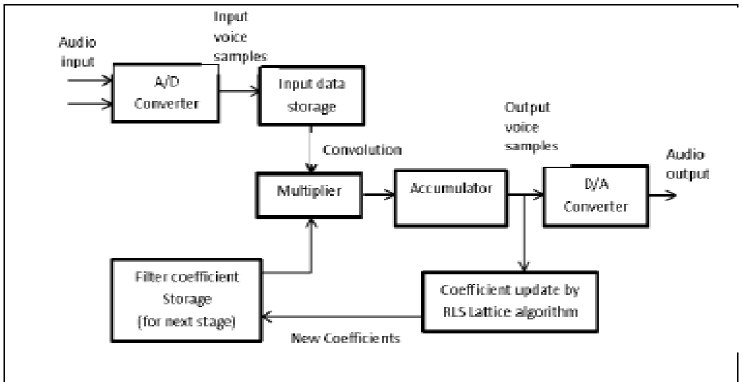

Fig. 2 Block Diagram Representation of ANC

3.1 Simulation of Sinusoidal Signal

The input signal: x(n)=sin(0.05*pi*n)

Noise: v(n)=randn(1,n)

Simulation:

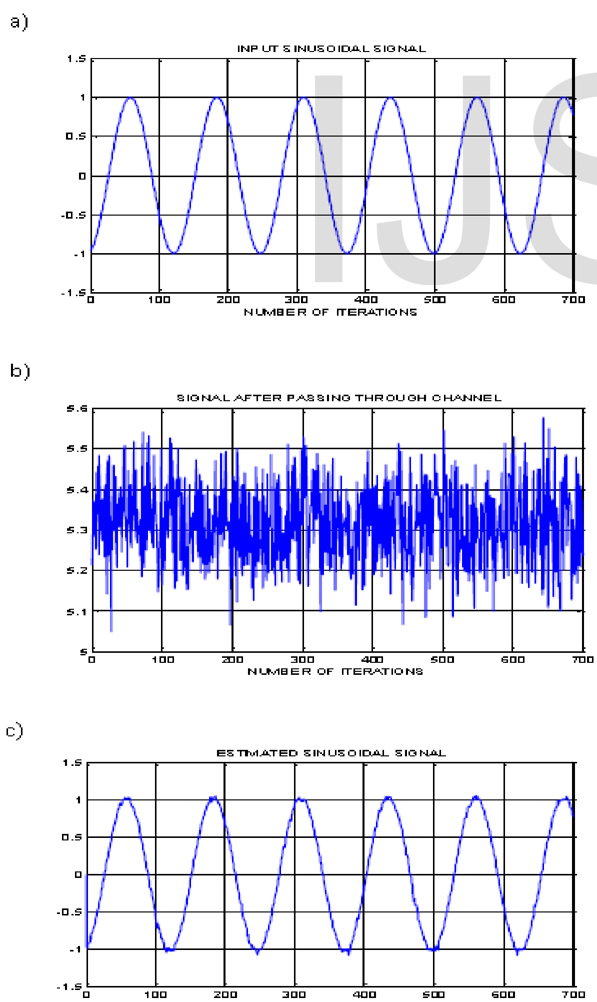

Fig. 3 Simulation of Sinusoidal Signal

(a)input sinusoidal signal (b) input signal mixed with noise (c)output of the RLSL filter

3.2 Simulation of Audio Signal

In this experiment the audio file is recorded from the sound card as a .wav file. This audio signal is then mixed with the white Gaussian noise and then passed through the RLSL algorithm based filter to get the desired output. The RLSL filter updates its coefficients until the nose from the mixed signal is removed.

Simulation:

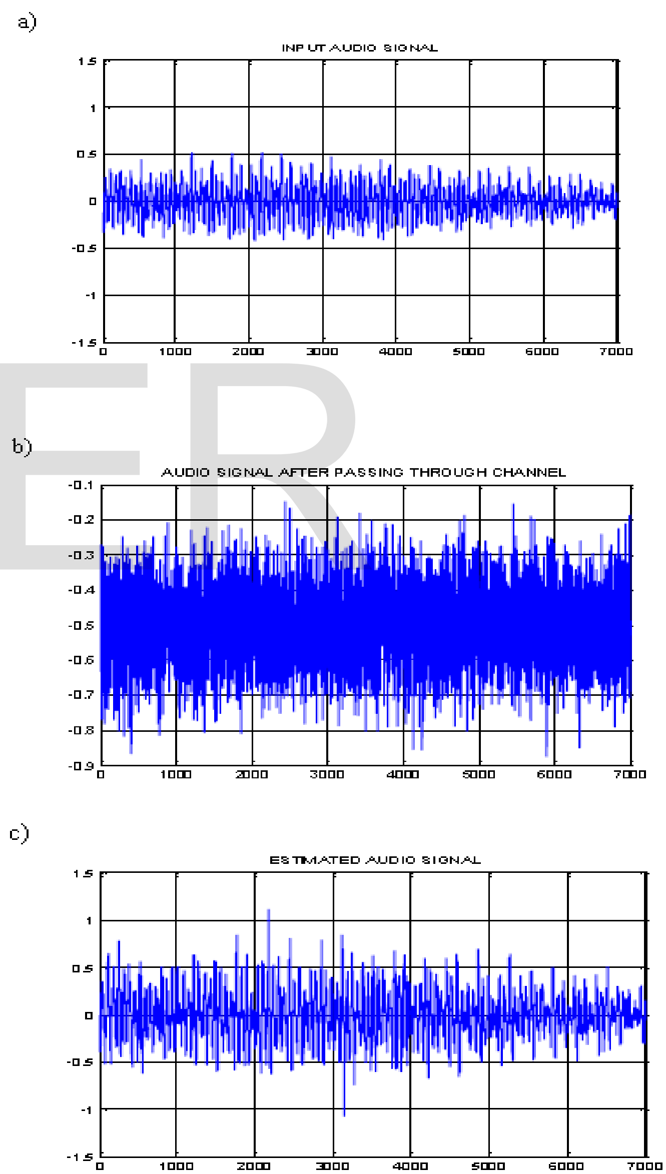

Fig. 4 Simulation of Audio Signal
(a)input audio signal (b) input signal mixed with noise (c)output of the RLSL filter.

d)

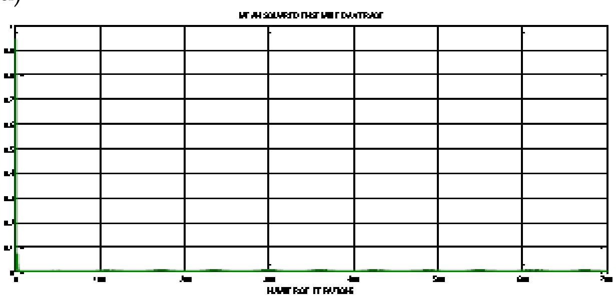

Fig. 5. Mean square Ensemble Average

The RLSL gives exact convergence as demonsrated in Fig. 5, and hence the output of the RLSL filter is exaclty match with the desired signal. The SNR (dB) calculated both at input and output of the RLSL filter is given below.

Input SNR (dB) = --24.1894
Output SNR (dB) = 27.1987

TABLE 1
THE STANDARD RLS LATTICE ALGORITHM

Initialization:

For $p = 0 \; to \; P$ set:

$$\Delta_p(-1) = 0$$
$$b_p(-1) = 0$$
$$\sigma^{f^2}_{p,-1} = 0$$
$$\sigma^{b^2}_{p,-1} = \delta; \text{ such that } \delta > 0 \;\&\; \delta \approx 0$$

For $n \geq 0$, repeat:

$$f_0(n) = b_0(n) = x(n)$$
$$\sigma^{f^2}_{0,n} = \sigma^{b^2}_{0,n} = \lambda \sigma^{f^2}_{0,n-1} + x^2(n)$$
$$\vartheta_0(n) = 1$$

For $p = 0 \; to \; P-1$, repeat:

$$\Delta_p(n) = \lambda \Delta_p(n-1) + \frac{f_p(n) b_p(n-1)}{\vartheta_p(n)}$$
$$f_{p+1}(n) = f_p(n) - \frac{\Delta_{p,n}}{\sigma^{b^2}_{p,n-1}} b_p(n-1)$$
$$b_{p+1}(n) = b_p(n-1) - \frac{\Delta_{p,n}}{\sigma^{f^2}_{p,n}} f_p(n)$$
$$\vartheta_{p+1}(n) = \vartheta_p(n) + \frac{b_p^2(n-1)}{\sigma^{f^2}_{p,n}}$$
$$\sigma^{f^2}_{p+1,n} = \sigma^{f^2}_{p,n} - \frac{\Delta^2_{p,n}}{\sigma^{b^2}_{p,n-1}}$$
$$\sigma^{b^2}_{p+1,n} = \sigma^{b^2}_{p,n-1} - \frac{\Delta^2_{p,n}}{\sigma^{f^2}_{p,n}}$$

4 Conclusion

In this work, we developed lattice form of RLS adaptive filter based on the vector space treatment. The main conclusion is that the standard lattice recursions of Table 1 represent the only numerically reliable algorithm. The algorithm demonstrates the importance of using the newly updated iterations for the sensitivity of the noise estimation algorithm performance by using the forgetting factor in the algorithm which has been verified by several simulations to validate the algorithm in finite precision.

Acknowledgement

The authors are thankful to the incessant support and motivation from the faculty members of Electronics and Communication Engineering at NIT Jamshedpur. The contribution and guidance provided by the Supervisor needs a special mention.

References

[1] Simon Haykin, "Order-recursive adaptive filters," in *Adaptive Fiter Theory*, 4th ed., India, pp. 536-563.

[2] B.Farhang-Boroujeny, "Lattice filters" in *Adaptive Filters*, USA, pp. 357-410.

[3] Glenn E. Johnson, Robert A. Muir, Joseph M. Scanlan, and William M. Steedly, "Practical Comparison of Adaptive Filter Algorithms" Proc. IEEE, pp. 259-263, 1995.

[4] J.R. Bunch, R.C.Le Borne and I.K. Proudler, "Tracking ill-conditioning for the RLS-lattice algorithms" IEE Proc.-Vis. Image Signal Process., Vol. 145, pp. 1-5 , February 1998.

[5] R. B. Wallace and R. A. Goubran, "Noise Cancellation Using Parallel Adaptive Filters" IEEE transactions on circuits and systems-II: Analog and Digital signal processing, vol. 39, pp. 239-242, April 1992.

[6] Md. Zulfiquar Ali Bhotto and Andreas Antoniou "Robust Recursive Least-Squares Adaptive-Filtering for Impulsive- Noise Environments" IEEE Signal processing letters, vol. 18, pp. 185-188, March 2011.

[7] Ma Shengqian, Xu Guowei, Ma Zhifeng, Wei Shuping, Fan Manhong, "Research on Adaptive Noise Canceller of an Improvement LMS algorithm" Proc. IEEE, pp. 1611-1614, 2011.

Kumar Shashi Kant received his B.E in Electronics & Communication Engineering from Rajiv Gandhi Proudyogiki Vishwavidyalaya, Bhopal and is currently pursuing his M.Tech degree in VLSI Design & Embedded System at National Institute of Technology, Jamshedpur, India.His present research area is Signal Processing & VLSI Design.

Amit Prakash received his M. Tech. from N.I.T Jamshedpur, India and is pursuing his Ph. D in Electronics & Communication Engg. Presentl he is working as an Associate Professor in Department of Electronics & Communication Engineering at National Institute of Technology, Jamshedpur, India. His present area of interest is Signa Processing & VLSI Design.