American Journal of Engineering Research (AJER)
e-ISSN : 2320-0847 p-ISSN : 2320-0936
Volume-03, Issue-08, pp-212-226
www.ajer.org

Research Paper Open Access

Denoising Of TDM Signal Using Novel Time Domain and Transform Domain Multirate Adaptive Algorithms

C Mohan Rao[1], Dr. B Stephen Charles[2], Dr. M N Giri Prasad[3]
[1]*Asso. Prof., Department of ECE, NBKRIST, Vidhyanagar, India*
[2]*Principal, SSCET, Kurnool, India*
[3]*Prof. & Head, Department of ECE, JNTUACE, Anantapuram, India*

ABSTRACT : *The Gaussian noise is obvious in most of the communication channels. The impulsive noise tends to Gaussian form as the time slot extends over a significant period. In this paper a channel with Gaussian noise is considered. A TDM signal passed through this kind of channel was applied to denoising. The denoising was implemented in time as well as transform domain. The adaptive algorithms like LMS, NLMS, RLS, LSL and transform domain LMS implementations using wavelet packets are designed. Two new algorithms are proposed and verified with the above application; one a variation of LMS and second binary step size LMS. From the simulation results it was found that the multirate adaptive algorithms proposed give better performance than the existing techniques.*

KEYWORDS :*Denoising, TDM, LMS, Wavelet Packet*

I. LMS ALGORITHM

The simplified block diagram of a transversal adaptive FIR filter is depicted in figure 1, where the block denoted by adaptive filter comprises as adaptive filter $\hat{h}(n) = [\hat{h}_1(n), \hat{h}_2(n), ..., \hat{h}_N(n)]^t$ and algorithm, x(n) is the input sequence from which the input vector $X(n) = [x(n), x(n-1), ..., x(n-N+1)]^t$ is obtained, e(n) is the output error, $\hat{y}(n)$ is the output of the adaptive filter and d(n) is the desired signal. All the theoretical derivations are referred to figure 1.

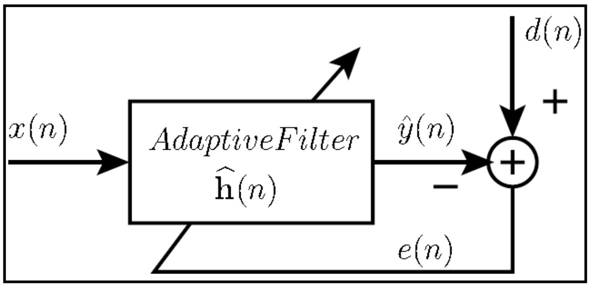

Fig. 1 Simplified block diagram of an adaptive FIR filter

In connection with figure 1, the output of the adaptive filter can be written as follows:

$$\hat{y}(n) = \hat{h}^t(n) X(n) = X^t(n)\hat{h}(n) = \sum_{i=1}^{N} \hat{h}_i(n) x(n-i+1) \quad (1)$$

wheret is the transposition operator. The output error is expressed by the following equation [1][2]:

$$e(n) = d(n) - \hat{y}(n) \quad (2)$$

The coefficients of the adaptive filter are updated to minimize the output mean squared error defined as follows:

$$J(n) = E\left[e^2(n)\right] = E\left\{[d(n) - \hat{y}(n)]^2\right\} \quad (3)$$

The optimum filter coefficients in the mean square sense are those coefficients for which the partial derivatives of *J(n)* equals to zero. Denoting the vector of the optimum coefficients as $h_o = \left[h_{o1},\ldots, h_{oN}\right]^t$, the system of equations which gives h_o is obtained as in the sequel:

$$\frac{\delta J(n)}{\delta h_{oi}} = \frac{\delta E\left\{[d(n) - y(n)]^2\right\}}{\delta h_{oi}}$$

$$= -2E\{x(n-i+1)[d(n) - y_o(n)]\} \quad (4)$$

$$= -2E\{x(n-i+1)e_o(n)\} = 0, \qquad \forall_i = 1,\ldots, N$$

where $e_o(n) = d(n) - h_o^t X(n)$ (5)

is the minimum output error obtained when the coefficients of the adaptive filter equals the coefficients of the optimum Wiener filter. Equation (4) can be written in a more compact form as follows:

$$E[X(n)e_o(n)] = 0 \quad (6)$$

It follows from (6) that the optimum error is orthogonal to the input vector at each time instant n, and this represents the well-known principle of orthogonality. From equation (4), the Wiener-Hopf equations which give the coefficients of the optimum filter are represented by:

$$\begin{cases} h_{o1}r(1-1) + h_{o2}r(1-2) + \ldots + h_{oN}r(1-N) = p(0) \\ h_{o1}r(2-1) + h_{o2}r(2-2) + \ldots + h_{oN}r(2-N) = p(1) \\ \ldots\ldots \\ h_{o1}r(N-1) + h_{o2}r(N-2) + \ldots + h_{oN}r(N-N) = p(N) \end{cases} \Leftrightarrow Rh_o = p \quad (7)$$

where $r(i-j) = E[x(n-i)x(n-j)]$,

$p(i) = E[d(n)x(n-i)]$

and $p = [p(1), p(2),\ldots p(N)]^t$

We note that the terms $r(i-j) = r(j-i)$ and $r(i-i) = r(j-j) = r(0)\ \forall\ i, j$, therefore the matrix **R** can be written as:

$$\mathbf{R} = \begin{bmatrix} r(0) & r(1) & r(2) & \ldots & r(N-1) \\ r(1) & r(0) & r(1) & \ldots & r(N-2) \\ . & . & . & & . \\ . & . & . & & . \\ . & . & . & & . \\ . & . & . & & . \\ r(N-1) & r(N-2) & r(N-3) & \ldots & r(0) \end{bmatrix} \quad (8)$$

When the matrix **R** is invertible and its elements can be estimated, the optimum Wiener filter can be easily obtained from (7) as:

$$h_o = \mathbf{R}^{-1} p \quad (9)$$

In situations when the elements of the matrix **R** are not available an iterative algorithm can be applied to the adaptive filter which transforms its coefficients toward h_o. One simple adaptive algorithm is the Steepest Descent (SD) algorithm, which updates the coefficients of the adaptive filter at each iteration in the opposite direction of the cost function gradient. In the case of the SD, the update formula for the filter coefficients is:

$$\hat{h}(n+1) = \hat{h}(n) - \frac{1}{2}\mu\nabla J(n) \quad (10)$$

where $\nabla J(n) = \left[\frac{\delta J(n)}{\delta \hat{h}_1(n)}, \frac{\delta J(n)}{\delta \hat{h}_2(n)}, ..., \frac{\delta J(n)}{\delta \hat{h}_N(n)} \right]^t$ and

$$\frac{\delta J(n)}{\delta \hat{h}_i(n)} = -2E[x(n-i+1)e(n)] \quad (11)$$

In order to compute the elements of the gradient in equation (11), the expectation operator must be used. A simpler alternative is to use the instantaneous gradient instead of the true gradient and the obtained algorithm is called the Least Mean Square (LMS). As a consequence, the LMS algorithm uses the following coefficient update formula:

$$\hat{h}(n+1) = \hat{h}(n) + \mu e(n)X(n) \quad (12)$$

where the step-size μ was introduced to control the stability of the algorithm. Finally, the LMS algorithm can be described by the following four steps:

1. From the input vector $X(n) = [x(n), x(n-1), ..., x(n-N+1)]^t$ from the input sequence x(n).
2. Compute the output of the adaptive filter: $\hat{y}(n) = X^t(n)\hat{h}(n) = \hat{h}^t(n)X(n)$.
3. Compute the output error: $e(n) = d(n) - \hat{y}(n)$.
4. Update the coefficients of the adaptive filter: $\hat{h}(n+1) = \hat{h}(n) + \mu e(n)X(n)$.

II. VARIATION OF LMS

If it were possible to make exact measurements of the gradient vector in all iterations and if the step-size parameter μ is suitably chosen, then the tap-weight vector computed by using the method of steepest-descent would indeed converge to the optimum Wiener solution. In reality, however, exact measurements of the gradient vector are not possible, and it must be estimated from the available data. In other words, the tap-weight vector is updated in accordance with an algorithm that adapts to the incoming data. One such algorithm is the least mean square algorithm. A significant feature of LMS is its simplicity; it does not require measurements of the pertinent correlation functions, nor does it require matrix inversion. The LMS algorithm is a search algorithm in which a simplification of the gradient vector computation is made possible by appropriately modifying the objective function. The LMS algorithm, as well as others related to it, is widely used in various applications of adaptive filtering due to its computational simplicity. The convergence speed of the LMS is shown to be dependent on the eigenvalue spread of the input signal correlation matrix. The LMS algorithm is by far the most widely used algorithm in adaptive filtering for several reasons. The main features that attracted the use of the LMS algorithm are low computational complexity, proof of convergence in stationary environment, unbiased convergence in the mean to the Wiener solution, and stable behavior when implemented with finite-precision arithmetic. Let x(n) and d(n) represent the reference input and the desired output signal, respectively, to the adaptive filter. Let L denote the total number of filter coefficients. Define the *L* x 1 coefficient vector H(n) and the input vector X(n) as

$$H(n) = [h_o(n), h_1(n), ..., h_{L-1}(n)]^T \quad (13)$$
$$X(n) = [x(n), x(n-l), \ldots, x(n-L+l)]^T \quad (14)$$

The LMS is described as

$$e(n) = d(n) - H^T(n)X(n) \quad (15)$$
$$H(n+1) = H(n) + \mu_s X(n)e(n) \quad (16)$$

In practice, (16) may be replaced with

$$H(n+1) = H(n) + \frac{\mu}{X^T(n)X(n) + \sigma} X(n)e(n) \quad (17)$$

or

$$H(n+1) = H(n) + \frac{\mu}{Lr(0)} X(n)e(n) \quad (18)$$

where the positive step-size μ is bounded by 2, σ is a small positive number and r(0)is the estimated autocorrelation function value of x(n) for lag 0.

Digital filter is the basic building block of Digital Signal Processing systems. Finite Impulse Response is preferred when compared to Infinite Impulse Response, because of its properties like guaranteed stability, linear phase and low response, but with expense of large number of arithmetic operations are involved. In communication systems channel noise and Inter Symbol Interference degrades the performance of communication system. To reduce this problem adaptive equalizers are used to shape the signals at the receivers. Least Mean Square technique is the one of the adaptive techniques. It is easy to realize, the computational complexity causes a long output delay, which is not tolerable. This computational complexity can be reduced using frequency domain adaptive filtering, but nonlinear systems performance degrades drastically. To overcome this, problem of derivative base and derivative free learning algorithm, we use natural selection or derivative free algorithms. A new evolutionary computation algorithm based on natural learning to update the weights of adaptive filter was used. In this method Genetic Algorithm (GA) is used to update weights of filter coefficients.

III. BINARY STEP-SIZE LMS

In practical applications adaptive algorithms which possess high convergence speed while maintaining small convergence error rate are of great interest. For instance, in channel equalization during the transient period, the frequency characteristic of the adaptive equalizer is far from the inverse of the frequency response of the channel therefore the data transmitted during this time will be corrupted. In echo cancellation application, if the coefficients of the adaptive canceler are not close to the coefficients of the FIR filter which models the echo path the resulting echo signal is not attenuated. Actually, it is possible in this application, that during the transient period, the echo will be actually amplified. As a consequence, the transient period of the adaptive filter must be as small as possible for most of the practical applications in order to improve the overall quality of the system. The LMS algorithm has a small computational complexity therefore; it is very simple to be implemented in practice. Although it is simplicity, one of its main drawbacks is the fact that the speed of convergence and steady state error depends on the same parameter, the step-size μ.

In conclusion, when a constant step-size is used in LMS, there is a tradeoff between the steady-state error and the convergence speed, which prevent a fast convergence when the step-size is chosen to be small for small output error. In order to deal with this problem, a simple idea is to use a step-size which is time-varying during the adaption. At early stages of the adaption, when the adaptive filter is far from the optimum, a larger value of the step-size should be used. This will shorten the transient period and increase the convergence speed of the adaptive filter. As the adaptive filter goes close to the optimum Wiener solution, the step-size should be decreased and so the misadjustment. The adaptive algorithms derived from the LMS, which uses time-varying step-size modified as described above, belong to the class of variable step-size LMS algorithms. The variable step-size LMS algorithm first introduced by Kwong and Johnston in [4] uses the following update formula for the adaptive filter coefficients:

$$\hat{h}(n+1) = \hat{h}(n) + \mu(n)e(n)x(n) \quad (19)$$

where $\hat{h}(n)$ is the N x 1 vector of the adaptive filter coefficients, x(n) is the vector of the past N samples from the input sequence, μ(n) is a time-varying step-size and e(n) is the output error.The time-varying step-size is also adapted as in the following equation:

$$\mu'(n+1) = \alpha\mu'(n) + \gamma e^2(n),$$

$$\mu(n+1) = \begin{cases} \mu_{max} & if \quad \mu'(n+1) > \mu_{max} \\ \mu_{min} & if \quad \mu'(n+1) < \mu_{min} \\ \mu'(n+1) & otherwise \end{cases} \quad (20)$$

with $0 < \alpha < 1$ and $\gamma > 0$ being some constant parameters and μ_{max} and μ_{min} being the upper and lower bounds of the time-varying step-size. The constant parameter μ_{max} which is normally selected close to the instability point of the conventional LMS algorithm is used to increase the convergence speed, while the parameter μ_{min} is chosen provide a good compromise between the steady-state misadjustment and the tracking capacity of the algorithm. The parameter γ is used to control the convergence time and also the steady-state level of the misadjustment. The behavior of the step-size as described in (20) is the following: at early stages of the adaption the step-size is increased due to the large value of the output error. As the algorithm goes closer to the steady-state the value of e(n) decreases which decrease the step-size μ_n.The following approximate analytical expression for the steady-state misadjustment of the variable step-size LMS algorithm was derived in [4]:

$$M = \frac{J_{ex}}{J_{\min}} = \frac{1 - \sqrt{1 - 2\frac{(3-\alpha)\gamma J_{\min}}{1-\alpha^2} tr[\mathbf{R}]}}{1 + \sqrt{1 - 2\frac{(3-\alpha)\gamma J_{\min}}{1-\alpha^2} tr[\mathbf{R}]}} \quad (21)$$

Clearly, the steady-state misadjustment depends on the parameter γ and on the minimum value of the MSE $J_{\min}$. Since the speed of convergence of the algorithm depends also on the parameter γ, one can conclude that there is still dependence between the misadjustment and the convergence time [5]. Another drawback of this algorithm is the fact that the steady-state misadjustment depends also on $J_{\min}$. For instance in system identification applications, the minimum MSE equals the output noise variance, therefore the steady-state misadjustment depends on the system noise.In this section, a new variable step-size algorithm is presented in which the step-size varies in two ways thereby naming binary step-size variation algorithm. Supervised channel equalization is considered [6]. Depending on the error the step-size gets updated. The updation process is shown in the figure 2.

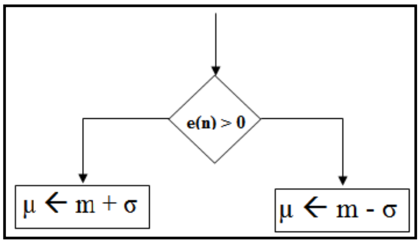

Fig. 2Updation process

In another observation it has been identified that if with the NLMS the maximum change of step-size is limited to 0.01, the convergence speed is very high [7].

IV. TRANSFORM DOMAIN LMS

The transform-domain LMS algorithm is another technique to increase the convergence speed of the LMS algorithm when the input signal is highly correlated. The basic idea behind this methodology is to modify the input signal to be applied to the adaptive filter such that the conditioning number of the corresponding correlation matrix is improved. In the transform-domain LMS algorithm, the input signal vector $\mathbf{x}(k)$ is transformed in a more convenient vector $\mathbf{s}(k)$, by applying an orthonormal (or unitary) transform [4], i.e.,

$$\mathbf{s}(k) = \mathbf{T}\mathbf{x}(k) \quad (22)$$

where$\mathbf{T}\mathbf{T}^T = \mathbf{I}$. The MSE surface related to the direct-form implementation of the FIR adaptive filter can be described by$\xi(k) = \xi_{\min} + \Delta\mathbf{w}^T(k)\mathbf{R}\Delta\mathbf{w}(k)$ (23)

where$\Delta\mathbf{w}(k) = \mathbf{w}(k) - \mathbf{w}_o$. In the transform-domain case, the MSE surface becomes

$$\begin{aligned}\xi(k) &= \xi_{\min} + \Delta\hat{w}^T(k)\, E[s(k)s^T(k)]\Delta\hat{w}(k) \\ &= \xi_{\min} + \Delta\hat{w}^T(k)\, \mathbf{TRT}^T \Delta\hat{w}(k)\end{aligned} \quad (24)$$

where $\hat{w}(k)$ represents the adaptive coefficients of the transform-domain filter. Figure 3 depicts the transform-domain adaptive filter. It can be noticed that the eccentricity of the MSE surface remains unchanged by the application of the transformation, and, therefore, the eigenvalue spread is unaffected by the transformation [8]. As a consequence, no improvement in the convergence rate is expected to occur. The Transform-Domain LMS Algorithm is as follows:

Initialization

$$x(0) = \hat{w}(0) = [0\ 0 \ldots 0]^T$$

γ = small constant

$0 < \alpha \le 0.1$

Do for each x(k) and d(k) given for k≥0

s(k) = **T**x(k)

$$e(k) = d(k) - s^T(k)\hat{w}(k)$$

$$\hat{w}(k+1) = \hat{w}(k) + 2\mu\, e(k) \Sigma^{-2}(k) s(k)$$

In the literature, Karhunen-Lo`eve Transform (KLT) is used as unitary transform for the transform-domain adaptive filter. However, since the KLT is a function of the input signal, it cannot be efficiently computed in real time. An alternative is to choose a unitary transform that is close to the KLT of the particular input signal [9]. By close is meant that both transforms perform nearly the same rotation of the MSE surface. In any situation, the choice of an appropriate transform is not an easy task. Some guidelines can be given, such as (however these are just conventions not rules):

1. Since the KLT of a real signal is real, the chosen transform should be real for real input signals;
2. For speech signals the discrete-time cosine transform (DCT) is a good approximation for the KLT;
3. Transforms with fast algorithms should be given special attention. A number of real transforms such as DCT, discrete-time Hartley transform, and others, are available.

Most of them have fast algorithms or can be implemented in recursive frequency-domain format [10]. In particular, the outputs of the DCT are given by

$$s_o(k) = \frac{1}{\sqrt{N+1}} \sum_{i=0}^{N} x(k-l) \quad (25)$$

$$s_i(k) = \sqrt{\frac{2}{N+1}} \sum_{l=0}^{N} x(k-l) \cos\left[\pi i \frac{(2l+1)}{2(N+1)}\right] \quad (26)$$

For complex input signals, the discrete-time Fourier transform (DFT) is a natural choice due to its efficient implementations [11]-[14].

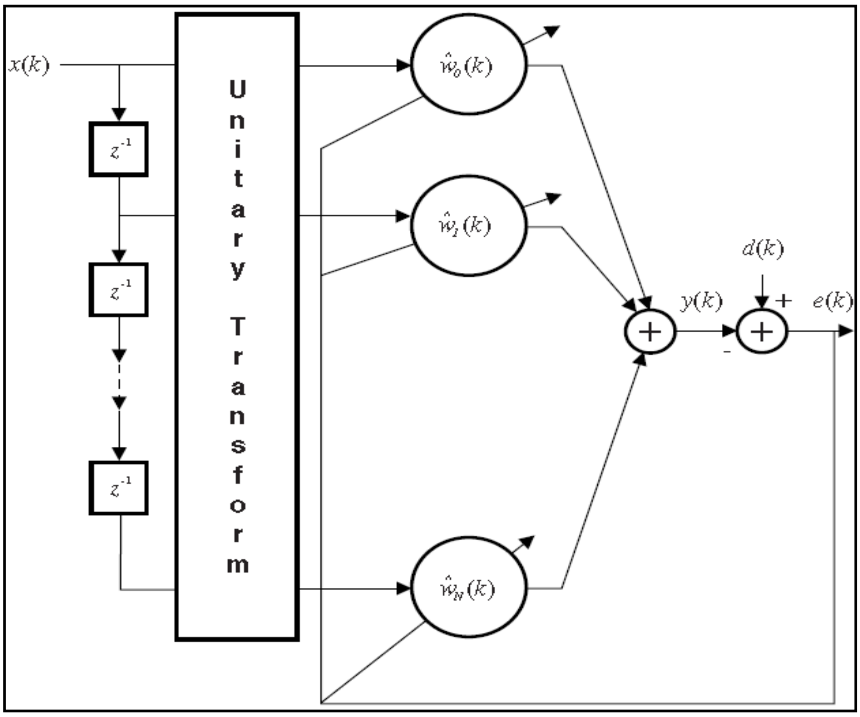

Fig. 3 Transform-domain adaptive filter

V. Wavelet Packet Transform

In this paper, the unitary transform used is wavelet packet transform. For almost all signals, the low-frequency component is the most important part. It is what gives the signal its significance and identity. The high-frequency content, on the other hand, adds flavor. Consider an audio signal.

If the high frequency components are removed, the audio sounds different, but one can still tell what's being said in the audio. However, if enough of the low-frequency components are removed, one hears gibberish.Wavelet analysis often speaks about approximations and details. The approximations are the low-frequency, high-scale components of the signal. The details are the high-frequency, low-scale components. The filtering process in wavelet analysis, at its basic level, looks something like figure 4.The original sequence, S, applied to two complementary filters and emerges as two signals as shown in figure 4. If a digital sequence of say 512 samples is applied to the filter bank consisting of one low and one high pass filter as mentioned above, the length of A will be 512 and that of D will also be 512. Hence the data to handle was doubled. But note that in A as well as in D only 256 samples are irredundant. To remove the redundant samples, the downsamplers are employed as shown in figure 5. The outputs are denoted by cA and cD.

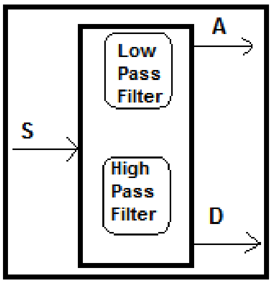

Fig. 4 Filtering Process in Wavelet Analysis

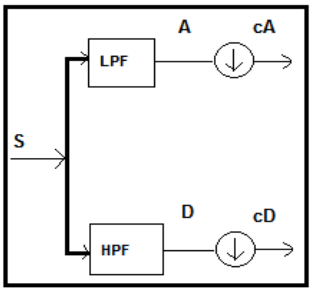

Fig. 5 Wavelet processing with downsamplers

This process, i.e., the conversion of S into cA and cD is called decomposition; the filters at this stage are referred as decomposition low pass and decomposition high pass filters. These filters have direct relation to the basis function used in a specific wavelet. The vectors cA and cD constitutes the DWT coefficients.

a. Multiple Stages of Decomposition

The decomposition process can be repeated means iterated, with successive approximations being decomposed in turn, so that one signal is broken down into many lower resolution components. This is called the wavelet decomposition tree shown in figure 6.

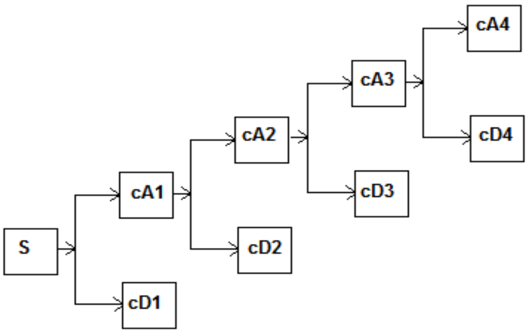

Fig. 6 Multistage Decomposition

The maximum number of decomposition stages should be taken so that the length of the sequence in the last stage is not less than 1. From the wavelet coefficients the original signal need to be recovered. The process of obtaining the original signal by using the wavelet coefficients is called reconstruction or synthesis shown in figure 7.The downsampling performed at decomposition stage introduces an aliasing effect. The reconstruction filters need to be selected so that the aliasing effect introduced at the decomposition stage should be cancelled. The overall process of wavelet is depicted in the figure 8.The wavelet packet analysis is an extension of wavelet analysis with an inclusion of analysis of both approximation (cA) and detail (cD) components. The wavelet packet analysis looks like a complete tree structure. The multistage wavelet packet analysis looks like as shown in figure 9.

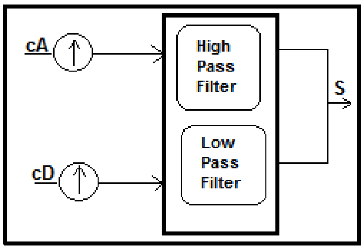

Fig. 7 Reconstruction Stage

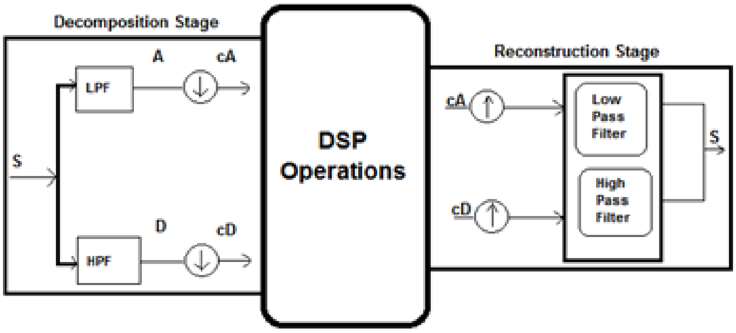

Fig. 8 Wavelet Transform as a mulirate filter

The wavelet packet analysis is an extension of wavelet analysis with an inclusion of analysis of both approximation (cA) and detail (cD) components. The wavelet packet analysis looks like a complete tree structure. The multistage wavelet packet analysis looks like as shown in figure 9.

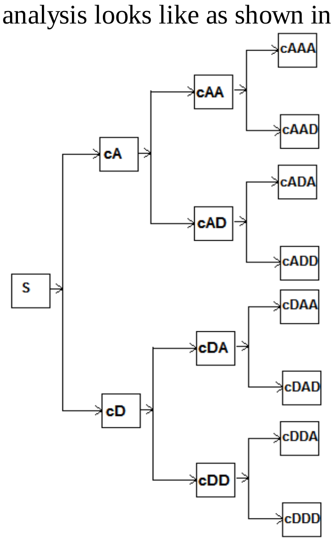

Fig. 9 Wavelet Packet Analysis

The wavelet packets use the wavelet filters to decompose and reconstruct the signals. The wavelet filters corresponds to the perfect reconstruction condition as well as to represent the data to suite different applications.

VI. SIMULATION RESULTS

In this section the simulation results of adaptive filters, Wavelets Packet based transform domain LMS, the variation of LMS (Var LMS), Binary step size LMS (BSS LMS), and multirateVar LMS and multirate BSS LMS in wavelet packet domain on de-noising of TDM signal are presented. The flow graph in figure 10 represents the total process that was considered in this paper.

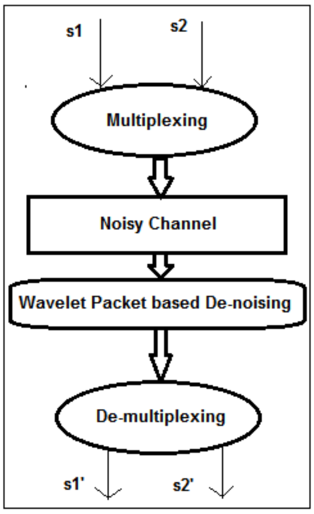

Fig. 10 De-noising of TDM signals based on Wavelet Packet based denoising

The two signals that are multiplexed in TDM, multiplexed signal and noisy signals are shown in the figure 11.

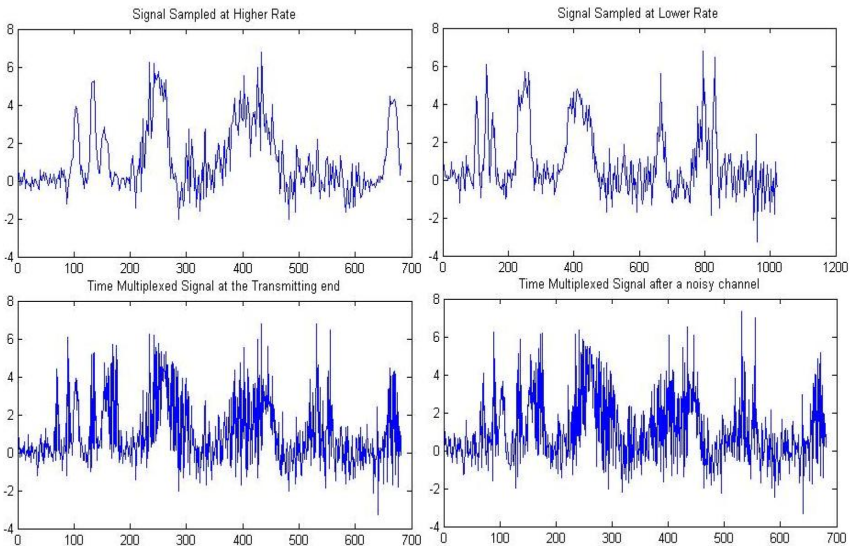

Fig. 11 Input, TDM and Noisy signals

The Peak Signal to Noise Ratio (PSNR), Mean Square Error (MSE), Maximum Squared Error (MAX ERR), Ratio of Squared Norms (L2RAT) are calculated and tabulated for different types of techniques.The simulation results of adaptive filters on denoising are presented in table I. LMS, Normalized LMS, RLS and LSL based denoising are implemented. It is observed that the maximum value of maximum error is 5.03 with LMS in the second signal case. The standard LMS has produced highest MSE and LSL a lowest MSE.

Table I: Denoising with Adaptive Filters

Denoising with Adaptive Filters								
	PSNR		MSE		MAX ERR		L2RAT	
	FS	SS	FS	SS	FS	SS	FS	SS
LMS	49.19	48.02	0.78	1.02	4.21	5.03	0.93	0.98
NLMS	52.30	52.32	0.38	0.38	2.04	3.17	0.95	0.93
RLS	53.37	52.39	0.30	0.38	1.93	2.74	0.98	1.03
LSL	54.02	53.87	0.26	0.27	1.56	2.13	0.91	0.96
Average	**52.22**	**51.65**	**0.43**	**0.51**	**2.44**	**3.27**	**0.95**	**0.97**

Table II presents the simulation results of denoising using LMS based on DCT and DFT. The performance of these techniques is almost similar to that of the previous adaptive filters except LMS. The standard LMS has shown fragile performance.

Table II: Denoising with Transform domain LMS based on DCT & DFT

Denoising with Transform domain LMS based on DCT & DFT								
	PSNR		MSE		MAX ERR		L2RAT	
	FS	SS	FS	SS	FS	SS	FS	SS
DFT	53.30	53.04	0.30	0.32	1.81	2.14	1.02	0.93
DCT	53.88	51.62	0.27	0.45	2.08	4.15	0.92	0.88
Average	**53.59**	**52.33**	**0.28**	**0.39**	**1.94**	**3.15**	**0.97**	**0.91**

Table III shows the performance of LMS in wavelet packet domain with different wavelets. Haar, Daubechies, Symlets, Coiflets, Biorthogonal, demeyer and reverse biorthogonal wavelets are considered. The performance with different wavelet combinations is almost similar. The PSNR is in the range of 54dB with almost all wavelets. The maximum error is around 1.6.

Table III: Denoising with LMS in Wavelet Packet Domain

Denoising with LMS in Wavelet Packet Domain								
	PSNR		MSE		MAX ERR		L2RAT	
	FS	SS	FS	SS	FS	SS	FS	SS
Haar	54.02	54.12	0.26	0.25	1.78	1.87	1.08	1.13
db10	54.51	53.76	0.23	0.27	1.33	1.86	1.06	1.06
db45	54.64	54.15	0.22	0.25	1.39	1.54	1.04	1.06
sym6	54.41	54.09	0.24	0.25	1.25	1.55	1.04	1.02
coif4	54.89	53.56	0.21	0.29	1.56	1.67	1.09	1.06
bior2.4	54.29	54.06	0.24	0.26	1.52	1.79	1.02	1.08
dmey	54.01	54.11	0.26	0.25	1.62	1.76	1.02	1.10
rbio1.3	54.39	54.05	0.24	0.26	1.70	1.78	1.04	1.06
Average	**54.40**	**53.99**	**0.24**	**0.26**	**1.52**	**1.73**	**1.05**	**1.07**

Table IV gives the simulation results of variation of LMS on denoising in five runs. The performance is almost similar to that of the previous techniques. The maximum value of PSNR is 54.08dB while the maximum error is 4.37.

Table IV: Denoising with Var LMS

Denoising with Var LMS								
	PSNR		MSE		MAX ERR		L2RAT	
	FS	SS	FS	SS	FS	SS	FS	SS
Var LMS-1	53.34	51.14	0.30	0.50	3.53	3.67	0.91	0.88
Var LMS-2	54.08	52.82	0.25	0.34	2.93	4.09	0.92	0.89
Var LMS-3	53.15	51.79	0.31	0.43	2.87	4.37	0.94	0.83
Var LMS-4	53.07	51.54	0.32	0.46	3.30	4.11	0.96	0.82
Var LMS-5	53.84	51.95	0.27	0.42	2.77	3.91	0.86	0.94
Average	**53.50**	**51.85**	**0.29**	**0.43**	**3.08**	**4.03**	**0.92**	**0.87**

Table V presents the performance of BSS LMS on denoising. The table gives the values of parameters in five runs. The average value of PSNR wit first signal is 53.07dB, with second signal 53.99dB, while the maximum error is 1.52 and 2.73 with two signals respectively. The performance of BSS LMS is found to be slightly better than the traditional LMS and transform domain LMS based on DCT and DFT and the variation of LMS, but the improvement is not significant.

Table V: Denoising with BSS LMS

Denoising with BSS LMS								
	PSNR		MSE		MAX ERR		L2RAT	
	FS	SS	FS	SS	FS	SS	FS	SS
BSS LMS-1	51.83	51.93	0.43	0.42	4.56	4.81	0.87	0.97
BSS LMS-2	54.22	52.11	0.25	0.40	2.52	3.70	0.89	0.89
BSS LMS-3	52.83	51.38	0.34	0.47	2.58	4.02	0.95	0.84
BSS LMS-4	53.33	52.11	0.30	0.40	2.99	3.92	0.91	0.86
BSS LMS-5	53.18	52.37	0.31	0.38	2.76	3.16	0.94	0.85
Average	**53.07**	**51.98**	**0.33**	**0.41**	**3.08**	**3.92**	**0.91**	**0.88**

Table VI presents the performance of the multirate variation of LMS by using wavelet packet domain on denoising. The simulation results show that the performance is improved by a great extent. The PSNR values at times 66.37dB and 64.46dB. The minimum value of PSNR is 58.86 with demeyer wavelet. The maximum error is in the range of 1.5. The results are obvious because of the correlated data coming out of wavelet packet representation.

Table VI: Denoising with MultirateVar LMS using Wavelet Packet Domain

Denoising with MultirateVar LMS using Wavelet Packet Domain								
	PSNR		MSE		MAX ERR		L2RAT	
	FS	SS	FS	SS	FS	SS	FS	SS
Haar	60.10	66.37	0.06	0.02	1.50	1.31	1.04	1.04
db10	59.11	61.16	0.08	0.05	1.60	1.57	1.06	1.06
db45	61.26	64.46	0.05	0.02	1.34	1.35	1.07	1.07
sym6	60.56	61.74	0.06	0.04	1.54	1.75	1.07	1.06
coif4	61.25	61.61	0.05	0.04	1.63	1.45	1.12	1.01
bior2.4	61.17	63.65	0.05	0.03	2.15	1.56	1.04	1.07
dmey	58.86	61.53	0.08	0.05	1.55	1.63	1.07	1.08
rbio1.3	62.28	61.21	0.04	0.05	1.33	1.52	1.04	1.05
Average	**60.58**	**62.72**	**0.06**	**0.04**	**1.58**	**1.52**	**1.06**	**1.06**

Table VII describes the performance of multirate BSS LMS by using wavelet packet domain on denoising. The table shows that the performance of BSS LMS in wavelet packet domain is even better than that of Var LMS. The average values of PSNR of Var LMS in wavelet packet domain is 60.58dB and 62.72dB with the two signals while that of BSS LMS is 63.55dB and 62.31dB. The maximum error and ratio of squared norms are similar with Var LMS and BSS LMS in wavelet packet domain.

Table VII: Denoising with Multirate BSS LMS using Wavelet Packet Domain

Denoising with Multirate BSS LMS using Wavelet Packet Domain								
	PSNR		MSE		MAX ERR		L2RAT	
	FS	SS	FS	SS	FS	SS	FS	SS
Haar	64.35	59.81	0.02	0.07	2.29	1.82	1.07	1.09
db10	66.74	63.15	0.01	0.03	1.19	1.41	1.08	1.06
db45	61.04	62.29	0.05	0.04	1.40	1.64	1.04	1.09
sym6	62.55	64.95	0.04	0.02	1.34	1.56	1.06	1.03
coif4	59.06	60.45	0.08	0.06	1.94	1.94	1.12	1.07
bior2.4	67.23	62.25	0.01	0.04	1.25	1.36	1.01	1.10
dmey	61.02	62.11	0.05	0.04	1.36	1.53	1.05	1.03
rbio1.3	66.42	63.47	0.01	0.03	1.54	1.32	1.05	1.09
Average	**63.55**	**62.31**	**0.04**	**0.04**	**1.54**	**1.57**	**1.06**	**1.07**

The comparison of all these techniques is presented in the table VIII and figures 12 to 15.

Table VIII: Denoising by different techniques

Denoising by different techniques								
	PSNR		MSE		MAX ERR		L2RAT	
	FS	SS	FS	FS	SS	FS	FS	SS
Adaptive Filter	52.68	51.88	0.38	0.47	2.27	3.23	0.95	0.95
DCT &DFT based LMS	53.59	52.33	0.28	0.39	1.94	3.15	0.97	0.91
BSS LMS	53.07	51.98	0.33	0.41	3.08	3.92	0.91	0.88
Var LMS	53.50	51.85	0.29	0.43	3.08	4.03	0.92	0.87
AdFl in WP	54.40	53.99	0.24	0.26	1.52	1.73	1.05	1.07
MultirateVar LMS WP	60.58	62.72	0.06	0.04	1.58	1.52	1.06	1.06
Multirate BSS LMS WP	63.55	62.31	0.04	0.04	1.54	1.57	1.06	1.07

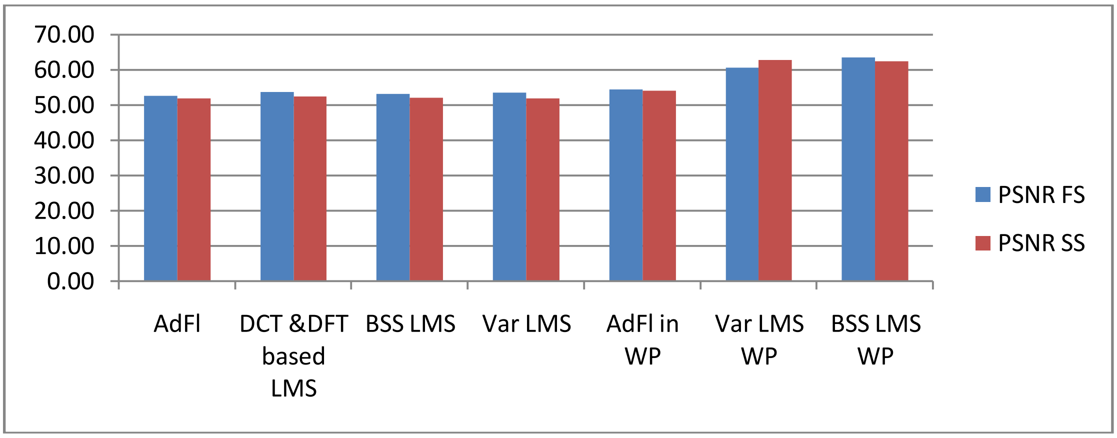

Fig. 12 PSNR values with different techniques

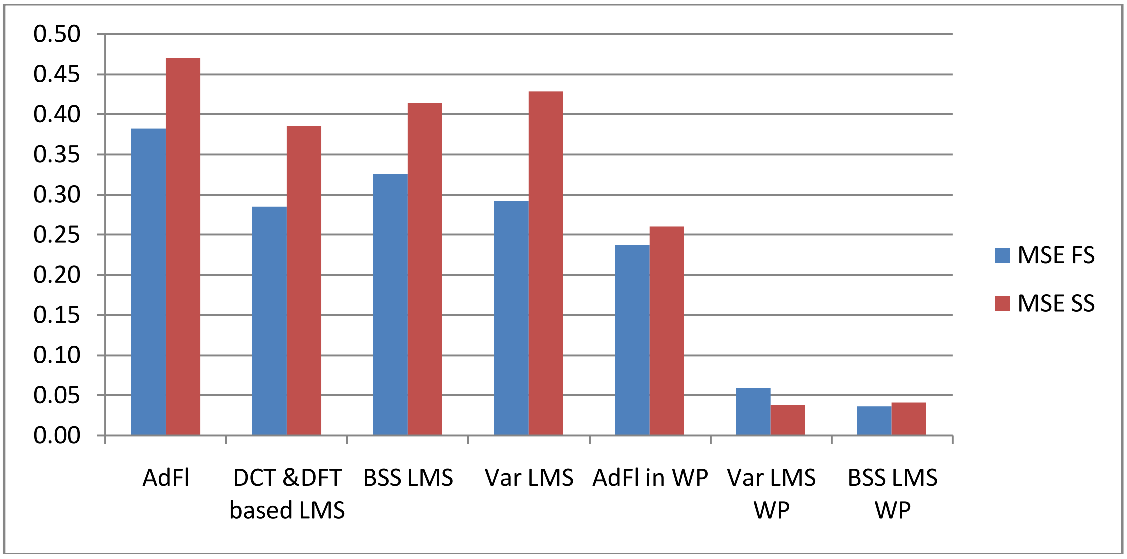

Fig. 13 MSE values with different techniques

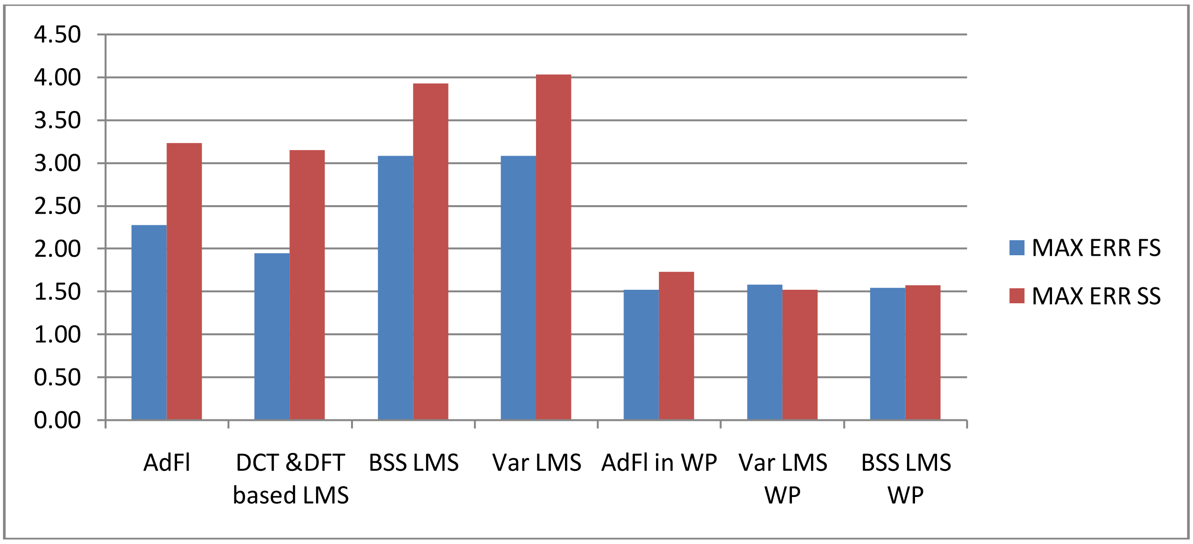

Fig. 14 Maximum error values with different techniques

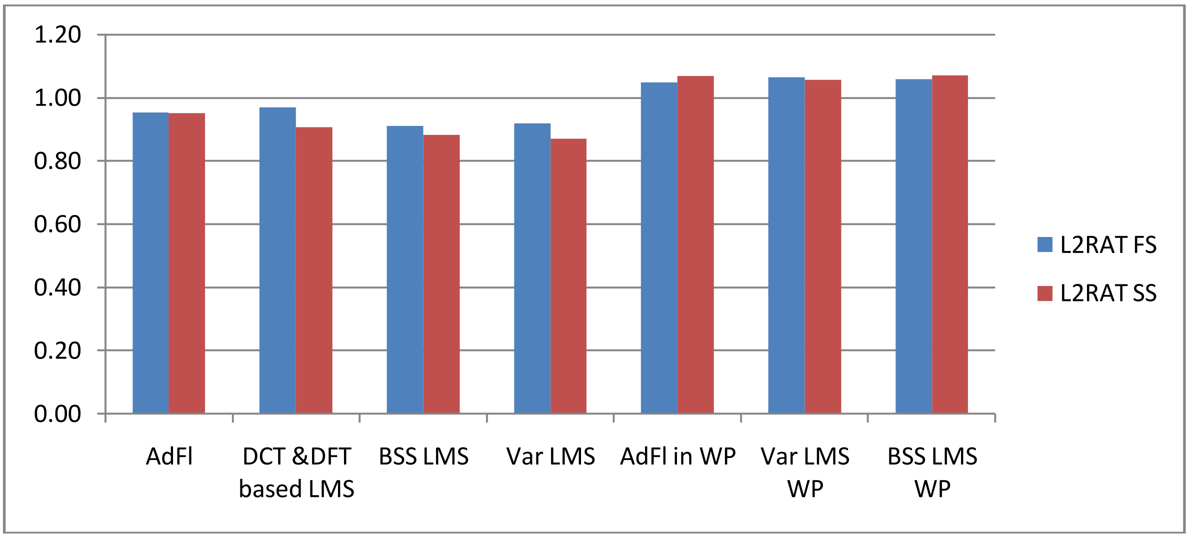

Fig. 15 L2RAT values with different techniques

VII. CONCLUSIONS

In this paper the denoising of time division multiplexed version of two signals sampled at different rate is considered. Traditional adaptive algorithms including DCT and DFT based transform domain LMS, transform domain LMS based on wavelet packets with different wavelets, a new variation of LMS, binary step size LMS and transform domain version of these two algorithms using wavelet packets with different wavelets are implemented for the specified denoising. The average PSNR with the traditional adaptive algorithms excluding DCT and DFT based transform domain LMS is calculated to be 52.28dB, with DCT and DFT based transform domain LMS 52.96dB, with BSS LMS 52.53dB, with Var LMS 52.67dB, with LMS in wavelet packet domain 54.19dB, Var LMS in wavelet packet domain 61.65dB and with BSS LMS in wavelet packet domain 62.93dB. From these results one can conclude that the new algorithms devised outperforms the existing techniques.

REFERENCES

[1] S.S. Hykin, “Adaptive Filter Theory”, NJ: Prentice – Hall, Englewood Cliffs, 1991.

[2] Dixit S., Nagaria D., “Neural network implementation of Least-Mean-Square adaptive noise cancellation”, IEEE International Conference on Issues and Challenges in Intelligent Computing Techniques (ICICT), pp. 134 – 139, Feb. 2014

[3] R.H. Kwong and E. W. Johnston, “A variable step size LMS algorithm”, IEEE transactions on signal processing, vol. 40, pp. 1633-1642, July 1992.

[4] J. C. M. Bermudez and N. J. Bershad, “A nonlinear analytical model for the quantized LMS algorithm: The arbitrary step size case,” IEEE Transactions on Signal Processing, vol. 44, pp. 1175-1183, May 1996.

[5] Borisagar.K.R, Kulkarni.G.R., “Simulation and Performance Analysis of Adaptive Filter in Real Time Noise over Conventional Fixed Filter”, IEEE International Conference on Communication Systems and Network Technologies (CSNT), pp. 621 – 624, May 2012.

[6] Haichen Zhao, Shaolu Hu, Linhua Li, Xiaobo Wan, “NLMS adaptive FIR filter design method”, TENCON 2013 - 2013 IEEE Region 10 Conference, pp. 1 – 5, Oct. 2013.

[7] Koike S, “Performance analysis of adaptive step-size least mean modulus-Newton algorithm for identification of non-stationary systems”, IEEE 8th International Colloquium on Signal Processing and its Applications (CSPA), pp. 184 – 187, March 2012

[8] Singh A, Bansal B.N., “Analysis of Adaptive LMS Filtering in Contrast to Multirate Filtering”, 4th International Conference on Emerging Trends in Engineering and Technology (ICETET), pp. 22 – 26, Nov. 2011.

[9] Yuan Feng, ZhihaiZhuo, Shengheng Liu, Xiaokang Cheng, Letao Zhang, Zhipeng Zhang, “A novel adaptive filter based on FDLMS and modified MFLMS”, IEEE International Conference on Information Science and Technology (ICIST), pp. 1326 – 1329, March 2013.

[10] Kusljevic M.D., Poljak P.D., “Simultaneous Reactive-Power and Frequency Estimations Using Simple Recursive WLS Algorithm and Adaptive Filtering”, IEEE Transactions on Instrumentation and Measurement, Vol. 60, Issue 12, pp. 3860 – 3867, Dec. 2011.

[11] Vityazev V.V., Linovich A.Y., “A subband equalizer with the flexible structure of the analysis/synthesis subsystem”, IEEE Region 8 International Conference on Computational Technologies in Electrical and Electronics Engineering (SIBIRCON), pp. 174 – 178, July 2010

[12] Ramya J.V.S.L, “Alias free sub-band adaptive filtering”, IEEE Conference on Recent Advances in Space Technology Services and Climate Change (RSTSCC), pp. 119 – 124, Nov. 2010.

[13] Yan Liu, MucongZheng, “Adaptive receiver for multirate CDMA systems on Kalman criterion”, IEEE International Conference on Computer Science and Service System (CSSS), pp. 2963 – 2965, June 2011

[14] P. S. R. Diniz, E. A. B. da Silva, and S. L. Netto, “Digital Signal Processing: System Analysis and Design”, Cambridge University Press, Cambridge, UK, 2002.

C. Mohan Rao (India) born on 21st April 1975, is pursuing Ph.D in Jawaharlal Nehru Technological University Anantapur. He received Master of Technology from Pondicherry Central University in the Year 1999, Bachelor of Technology form S.V University, Tirupati in the year 1997. He started his carrier as hardware Engineer in Hi-com technologies, during 1999 to 2001, after he worked as Assistant Professor in G. P. R. E. C during 2001 to 2007 and as Associate Professor in N. B. K. R. I. S. T. from 2007 to till date. He is a member of IETE and ISTE.

Dr. B. Stephen Charles (India) born on the 9th of August 1965. He received Ph.d degree in Electronics & Communication Engineering from Jawaharlal Nehru Technological University, Hyderabad in 2001. His area of interest is Digital signal Processing. He received his B.Tech degree in Electronics and Communication Engineering from Nagarjuna University, India in 1986. He started his carrier as Assistant professor in Karunya institute of technology during 1989 to 1993, later joined as Associate Professor in K. S. R. M. College of Engg. During 1993 to 2001 after that he worked as Principal of St. John's College of Engineering & Technology during 2001 to 2007 and now he is the Secretary, Correspondent and Principal in Stanley Stephen College of Engineering & Technology, Kurnool. He has 24 years of teaching and research experience. He published more than 40 research papers in national and international journals and more than 30 research papers in national and international conferences. He is a member of Institute of Engineers and ISTE.

Dr. M.N. Giri Prasad received his B.Tech degree from J.N.T University College of Engineering, Anantapur, Andhrapradesh, India in 1982. M.Tech degree from Sri Venkateshwara University, Tirupati, Andhra Pradesh, India in 1994 and Ph.D degree from J.N.T. University, Hyderabad, Andhra Pradesh, Indian in 2003. Presently he is working as a Professor in the Department of Electronics and Communication at J.N.T University College of Engineering Anantapur, Andhrapradesh, India. He has more than 25 years of teaching and research experience. He has published more than 50 papers in national and international journals and more than 30 research papers in national and international conferences. His research areas are Wireless Communications and Biomedical instrumentation, digital signal processing, VHDL coding and evolutionary computing. He is a member of ISTE, IE & NAFEN.