5.1 INTRODUCTION:

Noise estimation is the basic concept of Speech enhancement. Due to inaccurate noise estimation, most of the existing methods produce residual noise in the enhanced speech signal and most of the algorithms are not designed for non-stationary noise environments.

It is been observed in the literature survey that most of the enhancement techniques produce structural changes which can affect the reconstructed signal. Kalman filter has overcome this problem by estimating the initial samples and further it is observed that Kalman filter design does not have a specific way of determining the initial conditions.

Once the estimations are updated by adding new observations, the problem speculates and hence the noisy speech signal is not able to restore. Recent studies have also proved that, enhancement algorithms are not designed for a specific type of noise or nature of noise. Hence there is a large trade off between the manufacturers and the researchers.

Normally, Kalman filter works on the principle of Recursive process which uses all the series history with an advantage of estimating the stochastic path of the coefficients instead of a deterministic one. This will avoid the problem of solving the possible estimation cut when structural changes happen but this will give rise to a problem of suppressing noise at higher dBs and it has shown an impact on the Pitch and Formants of the speech signal which in turn reflected the Quality and Intelligibility of the speech signal.

Therefore in this thesis, it is focused on enhancing the compressed noisy speech signal which is a difficult task. Most of the researchers have concentrated on
enhancing the noisy speech signal but not on the “compressed noisy speech signal” and it is done at the receiver but not at the transmitter.

The main objectives of the thesis are to enhance the compressed noisy speech signal. So the research has started concentrating on the objective of achieving high perceived quality (how well a human perceives the audio signals) and to achieve high measured intelligibility (The message is understood) at low bit rate (bits per second of speech).

Since in this thesis, Lossy Compression technique called LPC coding has been adopted and hence there is every possibility of loss in quality. It is proposed to calculate the Spectral distortion (dB), Computational complexities (K Flops), Memory requirements (Floats), Signal to Noise Ratio (in dB), Pitch (in Hz), Formants (Hz) and Mean opinion Score (MOS) to measure the Quality and Intelligibility of the enhanced speech signal.

So in this process after focusing on the current available enhancement techniques, it is further decided to propose a new enhancement technique called Recursive Filter. The proposed Recursive filter is based on the concepts of Kalman filter. Recursive filter is designed with an estimator which can estimate recursively and updates each frame of the speech signal. Hence it can be concluded that recursive filter can track and estimate non-stationary noise.

The research include calculation of Spectral distortion (dB), Computational complexities (K Flops), Memory requirements (Floats), Signal to Noise Ratio (SNR), Pitch (in Hz), Formants (Hz) and Mean Opinion Score (MOS) to measure the Quality and Intelligibility of the enhanced speech signal corrupted by additive white Gaussian noise at different dB’s and further it is extended with different environmental Real world noises which have been used as the metric.

These results shows that proposed approach will provide enhanced speech with very less distortion when compared with Spectral Subtraction and Kalman filter methods using method-1 i.e., “First enhancement and then compression”.
5.2 CONCEPT OF RECURSIVE FILTER:

This recursive Filter is an estimator for what is called the “Linear Quadratic Problem”, which focuses on estimating the instantaneous “state” of a linear dynamic system perturbed by white noise [18]. Statistically, this estimator is optimal with respect to any quadratic function of estimation errors.

5.3 THE RECURSIVE PROCESS:

After going through some of the introduction and advantages of using this filter, it is to focus on the process. The process [18] starts with the problem of trying to estimate the state of a discrete-time controlled process which is ruled by the laws of linear stochastic difference equation:

\[ x_k = Ax_{k-1} + Bu_k + w_{k-1} \]  

(5.1)

Where, \( x_k \) is the current state estimate, \( A \) is the state Matrix of size \( n \times n \), \( B \) is control input matrix of size \( n \times 1 \) and \( w \) is the random variable representing the process noise. The actual measurement equation is given by

\[ z_k = Hx_k + v_k \]  

(5.2)

Where, \( v_k \) is the random variable representing measurement noise, \( z_k - Hx_k \) is the residual measurement and \( Hx_k \) is the predicted measurement of the state and if the residual measurement is zero then it means that predicted and actual are the same. Hence it can be confirmed that the values are close to each other and the error is close to zero.

Random variables are considered as independent and are in normal probability distributions and hence they get whiteness characteristics of probability distributed function.
During the theoretical process of measurement design, covariance vectors $Q$ and $R$ are assumed as constant and it is also observed in practice that, they change from time to time during measurement.

So, in the absence of either a driving function or process noise, matrix $A$ in the difference equation (5.1) relates the state at the previous time step $k - 1$ to the state at the current step $k$. In practice, matrix $A$ might change with each time step of $k$, however in this case, it is assumed constant.

In the above measurement equation (5.2), $H$ might change with each time step of $k$, however it is assumed as constant.

5.4 ALGORITHM APPROACH:

This section begins with a broad overview, covering the "high-level" operation of one form of this filter. After presenting this high-level view, the specific equations and their use in this discrete version of the filter are focused.

Firstly, it estimates a process by using a form of feedback control loop whereby the filter estimates the process state at some time and then obtains feedback in the form of (noisy) measurements. As such, these equations for this filter fall into two groups: “Time Update equations” and “Measurement Update equations”.

The responsibilities of the time update equations are for projecting forward (in time) the current state and error covariance estimates to obtain the priori estimates for the next time step. The measurement update equations are responsible for the feedback i.e. for incorporating a new measurement into the priori estimate to obtain an improved posteriori estimate.
The time update equations can also be thought of as “predictor” equations, while the measurement update equations can be thought of as “corrector” equations. By and large, this loop process of the final estimation algorithm resembles that of a predictor-corrector algorithm for solving numerical problems just like the one shown in Figure: 5.1 below.

![Figure: 5.1 Process of prediction and correction](image)

As the time update projects the current state estimate ahead in time, the measurement update adjusts the projected estimate from the time update by an actual measurement at that particular time. The specific equations for the “time” and “measurement” updates are presented below in Table 5.1 and Table 5.2

\[ x_k = Ax_{k-1} + BU_k \]  
\[ P_k = AP_{k-1}A^T + Q \]

From the Table 5.1, the time update equations in project its state, \( x \) and error covariance \( P_k \) estimates at the time step \( k-1 \) to step \( k \). As mentioned earlier, the matrices \( A \) and \( B \) are shown in equation (5.1), while \( Q \) is from (5.3). The process can be extended by taking the initial conditions of the discrete recursive filter update equations

\[ R_k = P_kH^T(HP_kH^T + R)^{-1} \]  
\[ X_k = x_k + (z_k - Hx_k) \]  
\[ P_k = (I - R_kH)P_k \]
As per the above equations it is clear that measurement update will start only when the recursive filter gain $R_k$ is computed. After comparing equation (5.7) with the equation presented in Table: 5.2 it is found that both are similar and with the previous section.

The next step is to measure the process in order to get $z_k$ so that, posteriori state estimate $X_k$ is obtained by incorporating the measurement in equation (5.8). At last posteriori error covariance estimate is obtained from equation (5.9).

From the previous discussions it is then concluded that the process will continue to predict the new time step priori estimates and update time and measurement pairs.

This Recursive nature is one which is best suitable for designing a filter that makes practical implementations much more feasible than earlier which were implemented with Kalman filter which has designed to operate on all of the data directly for each estimate.

Thus, the proposed recursive filter recursively conditions the current estimate on all of the past measurements.

The update equations are represented in Table 5.1 and Table 5.2, which gives much more complete and clear picture of the operation of the recursive filter.

**Time update(“predict”)**

1. Project the state head
   \[ \hat{x}_k^- = f(\hat{x}_{k-1}, u_k, 0) \]
2. Project the error covariance ahead
   \[ P_k^- = A_k P_{k-1} A_k^T + W_k Q_{k-1} W_k^T \]

Table 5.1: Time update equations
**5.5 IMPLEMENTATION:**

From a statistical point of view, many signals such as speech exhibit large amounts of correlation. From the perspective of coding or filtering, this correlation can be put to good use. The all pole, or autoregressive (AR), signal model is often used for speech. The AR signal model is introduced as:

\[
y^k = \frac{1}{1 - \sum_{i=1}^{N} \alpha_i Z_i} W_k
\]  

(5.10)

Equation (5.10) can also be rewritten in this form as shown below:

\[
y_k = a_1y_{k-1} + a_2y_{k-2} + \cdots + a_{N}y_{k-N} + W_k
\]  

(5.11)

Where,

- \( k \rightarrow \) Number of iterations;
- \( y_k \rightarrow \) Current input speech signal sample;
- \( y_{k-N} \rightarrow (N-1)\text{th} \) sample of speech signal;
- \( a_N \rightarrow N\text{th} \) filter coefficient;
- \( R_k \rightarrow \) Recursive filter gain; and
- \( W_k \rightarrow \) Excitation sequence (white noise).

In order to apply this filtering to the speech expression shown above, it must be expressed in state space form as

**Measurement update (“correct”)**

1. Compute the gain
   \[
   R_k = P_k^{-1} H_k^T (H_k P_k^{-1} H_k^T + V_k R_k V_k^T)^{-1}
   \]

2. Update estimate with measurement
   \[
   \hat{x}_k = \hat{x}_k^{-} + R_k (Z_k - h(\hat{x}_k^{-}, 0))
   \]

3. Update the error covariance
   \[
   P_k = (I - R_k H_k) P_k^{-}
   \]
\[ H_k = XH_{k-1} + \overline{W}_k \]  \hspace{1cm} (5.12)

\[ y_k = gh_k \]  \hspace{1cm} (5.13)

\[ X = \begin{pmatrix} a_1 & a_2 & \cdots & a_{N-1} & a_N \\ 1 & 0 & \cdots & 0 & 0 \\ 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & 0 \end{pmatrix} \]

\[ H_k = \begin{pmatrix} y_k \\ y_{k-1} \\ y_{k-2} \\ \vdots \\ y_{k-N+1} \end{pmatrix} \]

\[ w_k = \begin{pmatrix} w_k \\ 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \]

\[ g = (1 \ 0 \ \ldots \ 0) \]

From the above equation (5.12) \( X \) represents the system matrix, \( H_k \) represents a series of speech samples; \( \overline{W}_k \) is the excitation vector and \( g \) is the output vector.

\( H_k \) consists of \( N \) samples and the state vectors modifies from \( k-N+1 \)th iteration to the new state vector and from the \( k \)th iteration back to the \( (k-N+1) \)th iteration. The above equations will serve for the best iterations of this filter. As per the previous discussion, the filter equations specified in Table: 5.1 and Table: 5.2 functions in a looping method.

The following steps are involved within the looping process of the filter. So to start the process, let matrix \( H_{k-1}^T \) as the row vector and \( z_k = y_k \), then

\[ H_{k-1}^T = [y_{k-1} y_{k-2} \ \ldots \ \ldots \ \ldots \ \ldots \ y_{k-N}] \]  \hspace{1cm} (5.14)
Then equation (5.11) and (5.14) yield to

$$z_k = H_{k-1}^TX_k + W_k$$  \hspace{1cm} (5.15)

Let, $X_k$ will be continuously updated according to the number of iterations $k$.

Note that if the value of $k = 0$, the matrix $H_{k-1}$ is unable to be determine. However, it is already discussed above that, when the time $z_k$ is detected, the value in matrix $H_{k-1}$ will be known. So the above initializations and assumptions are sufficient enough for defining the recursive filter.

Which, consists of

$$X_k = [1 - R_kH_{k-1}^TX_{k-1} + R_kZ_k]$$ \hspace{1cm} (5.16)

Where;

$$I = \begin{bmatrix}
1 & 0 & \cdots & 0 & 0 \\
0 & 1 & \cdots & 0 & 0 \\
\vdots & \vdots & \ddots & \vdots & \vdots \\
0 & 0 & \cdots & 1 & 0 \\
0 & 0 & \cdots & 0 & 1
\end{bmatrix}$$

With

$$R_k = P_{k-1}H_{k-1}[H_{k-1}^TP_{k-1}H_{k-1} + R]$$ \hspace{1cm} (5.17)

Where, $R_k$ is the recursive filter gain.

$P_{k-1}$ is the prior error covariance matrix.

$R$ is the measurement noise covariance

$$P_k = P_{k-1} - P_{k-1}H_{k-1}[H_{k-1}^TP_{k-1}H_{k-1}]H_{k-1}^TP_{k-1} + Q$$ \hspace{1cm} (5.18)

Where $P_k$ is the posterior error co-variance Matrix and

$$Q = \begin{bmatrix}
1 & 0 & \cdots & 0 & 0 \\
0 & 1 & \cdots & 0 & 0 \\
\vdots & \vdots & \ddots & \vdots & \vdots \\
0 & 0 & \cdots & 1 & 0 \\
0 & 0 & \cdots & 0 & 1
\end{bmatrix}$$
Thereafter the reconstructed speech signal, $Y_k$ after filtering will be formed in a manner similar to (5.11)

$$Y_k = a_1 Y_{k-1} + a_2 Y_{k-2} + \ldots + a_N Y_{k-N} + W_k$$  \hspace{1cm} (5.19)

Therefore, $y_k$ is the input at the beginning of the recursive process and thereby there will not be any problem in deriving $H^T_{k-1}$. The process can continue in step of $k$ and therefore the final reconstructed speech signal $Y_k$ formed.

However he parameters $w_k$ and $\{a\}_{-1}$ are determined from application of this filter to the input speech signal $y_k$. So in order to construct $Y_k$, it is in need input matrix $X$ which contains the filtering coefficients and the white noise $w_k$, which both are obtained from the estimation of the input signal. This information is enough to determine $HH_{k-1}$

$$\text{Where, } HH_{k-1} = \begin{bmatrix} y_{k-1} \\ y_{k-2} \\ y_{k-N+1} \end{bmatrix}$$

Thus, forming the complete equation (5.19) as mentioned above.

The objective of the thesis is to first analyze the vector quantization techniques and come out with the quantization technique which will have less computational complexity and memory requirements, but it may contain more spectral distortion due to Lossy compression technique. So first the speech sample is recorded and additive noise like white Gaussian noise is added at various dBs right from 2 dB to 30 dB. These noisy speech signals are then compressed or enhanced first to get either compressed noisy speech signal or enhanced noisy speech signal. Vector quantization techniques like Unconstrained VQ and MSVQ are used to compress the speech signal and Spectral Subtraction method, Kalman filter method and proposed Recursive filter is used to enhance the noisy speech signals.

The above said procedures are also followed for speech signal corrupted by Real world noise signals and all the parameters are calculated and tabulated.

All the Simulations are done using MATLAB.