CHAPTER 2

DEVELOPMENT OF ALGORITHMS FOR DEMODULATOR

Satellite receiver processes signals that bear information as well as disturbances caused by the transmitter/receiver circuits and channel impairments such as fading and additive white Gaussian noise (AWGN). Usually, the receiver knows only some statistical properties of the signal and disturbances. From these statistical properties and using an observation over a finite interval of the received signal, the receiver is able to estimate the transmitted data symbols.

The receiver makes the decision on the received data using locally generated carrier oscillator and symbol clock, both of which are not referenced to the actual versions used to generate the data at the transmitter. The receiver has to estimate the offset between locally generated carrier and symbol clock to those used at the transmitter. Local carrier mismatches are labeled frequency offsets while clock mismatches are labeled symbol timing jitters or timing offsets.

Carrier frequency offset recovery is the process of estimating the offset between the frequency drift/change of the local oscillator and the actual carrier frequency transmitted. Symbol timing synchronization is the process in which the receiver estimates the offset between the locally generated symbol clock at the receiver and the actual symbol clock used at the transmitter. This offset estimate is used to sample the matched filter output at the correct timing instant.
that maximizes the signal to noise ratio. The receiver clock, when not matched to the transmitter clock, will cause the receiver symbol decision circuitry to sample the symbols at the wrong instance, resulting in detection errors. The problem of synchronization, thus, reduces to estimating the timing and frequency offsets in the carrier, using noisy samples of the received signal, under training.

2.1 CARRIER FREQUENCY OFFSET ESTIMATION

2.1.1 Review

Consider the general case of a passband signal which has been subjected to frequency offset. It can be expressed as

\[
\tilde{x}(t) = \text{Re}\left\{x(t)e^{j(\omega_c t + \theta(t))}\right\} \\
= \text{Re}\left\{[x_I(t) + jx_Q(t)]e^{j(\omega_c t + \theta(t))}\right\}
\]

(2.1)

Where \(\tilde{x}(t)\) is called the complex pre-envelope of the signal, \(x_I(t)\) and \(x_Q(t)\) are the in-phase and quadrature-phase components respectively, \(\omega_c\) is the carrier frequency in radians/second and \(\theta(t)\) models the frequency offset and/or phase jitter. If there is a constant frequency offset, then \(\theta(t)\) will have a linear term \(\omega_o t\). At the receiver, we demodulate this signal with the locally generated carrier \(e^{-j(\omega_c t + \phi(t))}\) where \(\phi(t)\) is the receiver’s estimate of the carrier phase. If, now, we sample this demodulated signal at the symbol rate, we get

\[
q_k = \text{Re}\left\{\tilde{x}(k)e^{j(\theta_k - \phi_k)}\right\}
\]

(2.2)
Where \( \theta_k \) and \( \phi_k \) are samples of \( \theta(t) \) and \( \phi(t) \) respectively. For constant \( \theta_k - \phi_k \) the received constellation will be a tilted version of the transmitted constellation as shown in Fig. 2.1(a). If the receiver demodulates with the wrong frequency \( \theta_k - \phi_k = \omega_0 k T \), where \( T \) is the symbol period, the received constellation rotates with an angular velocity of \( \omega_0 \) radian/sec., as shown in Fig. 2.1(b). If left uncorrected, the rotating constellation will make errors every time a received symbol rotates past the boundary of a decision region. To correct this, carrier offset estimation and subsequent compensation is needed. Two commonly used methods of frequency offset estimation at the receiver are data aided and non-data aided. We use data aided technique to recover the carrier offset in this work.

![Figure 2.1](image.png)

**Figure 2.1** Received Symbols when Receiver Demodulates (a) with Constant Phase Error and (b) with a Wrong Frequency.
2.1.2 Signal Model

The baseband equivalent of the received passband signal is given by

\[ \tilde{x}(t) = \sum_{n=-\infty}^{\infty} (a_n + jb_n) \bar{p}(t - nT + \tau) e^{j\theta(t)} + \tilde{n}_p(t) \]  

(2.3)

where \( \tau \) is the unknown timing offset, \( \theta(t) \) models unknown frequency offset, \( p(t) \) is the Root Raised Cosine (RRC) pulse and \( \tilde{n}_p(t) \) is the baseband equivalent of Additive White Gaussian Noise (AWGN). We demodulate \( \tilde{x}(t) \) with a local carrier \( e^{-j\phi(t)} \) where \( \phi(t) \) the receiver’s estimate of the carrier phase resulting in is

\[ \tilde{h}(t) = \sum_{n=-\infty}^{\infty} (a_n + jb_n) g(t - nT + \tau) e^{j\theta'(t)} + \tilde{n}_m(t) , \]  

(2.4)

Where \( \theta'(t) = \theta(t) - \phi(t) \)

For an ideal channel with AWGN, the optimum receiver filter is the “matched filter” that is matched to the transmitted pulse. Since the transmitter uses a Root Raised Cosine (RRC) pulse with 40% excess bandwidth as the shaping filter, the matched filter (MF) is also a RRC pulse with 40% excess bandwidth. The output of the matched filter is given by

\[ r(t) = \tilde{h}(t) \otimes p(t) \]

\[ = \sum_{n=-\infty}^{\infty} (a_n + jb_n) g(t - nT + \tau) e^{j\theta'(t)} + \tilde{h}(t) \]  

(2.5)

Where \( g(t) \) is the Raised Cosine (RC) pulse.

Satellite signal impairments are mostly due to the propagation channel effects and the transmitter/receiver circuitry of both the ground stations and the satellite transponder. Frequency offsets in satellite communication terminals are
experienced due to the factors such as oscillator frequency uncertainty, oscillator’s drift, and Doppler effects arising from vehicular motion with respect to the satellite. Depending on the carrier frequency and the relative velocity between the satellite and the ground receiver, frequency offsets can vary from few hundred Hz to several hundred KHz. Thus, we need to perform carrier frequency compensation. The timing recovery scheme that we will discuss in the later part of this Chapter requires a relatively crosstalk-free baseband signal. Hence we need to estimate the large frequency offset signal, compensate for it, then proceed for timing recovery and finally perform carrier phase tracking.

At the beginning of each packet, sixty-four (1, 1) symbols are used to estimate the carrier offset. In discrete time domain, the received signal for this training sequence is given by

\[ r(k) = \{a_n(k) + jb_n(k)\}e^{j\theta(k)} + \tilde{n}(k) \]  

(2.6)

With \( a_n(k) = b_n(k) = 1 \),

\[ r(k) = (1 + j1)e^{j\theta(k)} + \tilde{n}(k) \]

\[ = \sqrt{2} e^{j(\pi/4 + \theta'(k))} + \tilde{n}(k) \]

\[ \arg\{r(k)\} = \pi/4 + \theta'(k) + \text{some noise dependent term} \]

It can be noted that the phase of \( r(k) \), i.e., \( \pi/4 + \theta'(k) \) contains information about the frequency offset, but corrupted by noise. For constant phase offset, \( \theta'(k) = \theta_c \) (a constant) and for fixed frequency offset \( \omega_o, \theta'(k) = \omega_o k \). In both cases, \( \theta'(k) \) is a linear function of ‘\( k \)’. The problem, thus, reduces to fitting a straight line through the phase trajectory and finding its slope using Recursive Least Square Algorithm (RLS) [1].
2.1.3 Recursive Least Squares Based Estimation

The RLS algorithm can be used here after suitably modifying it, as explained in. [1]. Since RLS is a recursive algorithm, the slope is updated as each data point is available. Assume that we have points \((x_1, y_1), (x_2, y_2), (x_3, y_3), \ldots (x_i, y_i), (x_{i+1}, y_{i+1}), \ldots (x_N, y_N)\), through which we want to fit a line so as to minimize the sum of squares of errors. Considering observations in pairs, we generate two equations, i.e.,

\[
\begin{align*}
    y_1 &= m_1x_1 + c_1 \\
    y_2 &= m_1x_2 + c_1
\end{align*}
\]

Writing in matrix notation,

\[
Y_i = A_i Z_i
\]

where the index \(i\) corresponds to \(i^{th}\) iteration and

\[
Y_i = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix}, \quad A_i = \begin{bmatrix} x_1 & 1 \\ x_2 & 1 \end{bmatrix}, \quad Z_i = \begin{bmatrix} m_1 \\ c_1 \end{bmatrix}
\]

\(Z_i\) can be evaluated using the relation

\[
Z_i = Z_{i-1} + k_i(Y_i - A_i Z_{i-1})
\]

Where \(k_i = P_i A_i^T\) and \(P_i^{-1} = P_{i-1}^{-1} + A_i^T A_i\) \(\text{ (2.7)}\)

The inversion operation, in the above equation, is a computationally expensive task and can be avoided by using matrix inversion lemma given in Appendix A. Applying the matrix inversion lemma, \(P_i\) can be expressed as

\[
P_i = P_{i-1} - m_i A_i P_{i-1}
\]

where \(m_i\) is given by

\[
m_i = \frac{P_{i-1} A_i^T}{1 + A_i P_{i-1} A_i^T}
\]

Summarizing, the steps to be performed recursively to determine and y-intercept are
1. \[ m_i = \frac{P_{i-1} A_i^T}{1 + A_i P_{i-1} A_i^T} \]

2. \[ P_i = P_{i-1} - m_i A_i P_{i-1} \]

3. \[ k_i = P_i A_i^T \]

4. \[ Z_i = Z_{i-1} + k_i(Y_i - A_i Z_{i-1}) \]

In the next iteration, a new observation \((x_3, y_3)\) is obtained. We need to find the slope of the line that “passes” through \((x_1, y_1), (x_2, y_2)\) and \((x_3, y_3)\) with the criterion being that the sum of error squared is minimized. The previous estimates of slope and y-intercept need to be updated. We update the matrices \(Y_i, A_i\) and \(Z_i\) with

\[
Y_2 = \begin{bmatrix} y_2 \\ y_3 \end{bmatrix}, \quad A_2 = \begin{bmatrix} x_2 & 1 \\ x_3 & 1 \end{bmatrix}, \quad Z_2 = \begin{bmatrix} m_2 \\ c_2 \end{bmatrix}
\]

and algorithm proceeds as explained.

\((Y_i - A_i Z_{i-1})\) is called the Prediction Error and is a measure of the error during each iteration. These prediction errors are maintained and updated in every iteration. The squared prediction errors are to be minimized. The slope of the best fit line gives the estimate of frequency offset. Fig. 2.2(a) shows the phase of the samples in the absence of noise where Fig. 2.2(b) shows the observation points and the best fit line “passing through them” in the presence of noise.

If \(\hat{\theta}\) is the estimate of the carrier offset estimate, we correct the incoming signal by giving a rotation in the opposite direction, i.e., by multiplying the input sequence by \(e^{-j\hat{\theta}(k)}\). The resulting compensated stream has less crosstalk than the original uncompensated received signal. The timing recovery algorithm uses this corrected stream as its input. This method was seen to give a good estimate of the frequency offset in the presence of noise and cross-talk.
2.1.4 Unwrapping the angles

If the frequency offset $\theta'(k)$ is large, then the phase of $r(k)$ may exceed $2\pi$. Usually, any angle greater than $2\pi$ is represented as $\theta' \mod 2\pi$. This poses a problem during RLS implementation, since RLS works on unwrapped phase. One such sequence is shown in Fig. 2.3. If this sequence is the input to the RLS block, then erroneous results will be generated.

One approach to implement unwrapping is explained in. [2]. In this method, we need to compute the phase derivative and the principal value of phase at equally spaced frequencies. At each $\omega_k$, a phase estimate is performed by one step trapezoidal integration starting at $\omega_{k-1}$. If the difference between the estimate and the principal value is not within a given threshold (which can be specified), the step size is halved to get a new estimate.
Another way is to use discrete time approach to solve the wrap around problem as explained in [3]. The basic task in unwrapping is to distinguish a genuine transition of $2\pi$ radians and the spike introduced due to noise. To distinguish between the two, two absolute differences, a backward difference $b(n) = |\theta'(n) - \theta'(n-1)|$ and a forward difference $f(n) = |\theta'(n) - \theta'(n+1)|$ are maintained. To determine whether to add $2\pi$ radians to the phase or not, these two differences are compared with their respective thresholds (which are set by trial and error). In case of genuine $2\pi$ transition, the backward difference is large while the forward difference is small. In this case, $2\pi$ radians is added to the resolved angle. In case of noise, both the forward as well as the backward differences are large and hence, nothing is added to the resolved angle.

While this second technique gives satisfactory results in the absence of noise and in the presence of low noise, it fails to unwrap the phase satisfactorily.
Figure 2.4 Structure to Calculate the Phase Angle Between two Adjacent Samples

when the signal to noise ratio is small. We propose a technique that works well even at lower SNR. We correlate the current sample with the conjugate of the previous one as shown in Fig. 2.4. The probability of $r(k)r^*(k-1)$ exceeding $2\pi$ is very small. Therefore, phase unwrapping problem can be solved by accumulating these phases and then using RLS algorithm over it. It is seen that this above mentioned technique. Works with higher noise values as compared to the previously mentioned two unwrapping methods

2.2 TIMING RECOVERY

In this chapter, modified peak average energy criterion (PAEC) based timing recovery scheme [3] is presented.
2.2.1 Modified Peak Average Energy Criterion

Once we have estimated the bulk carrier offset and compensated the incoming signal for this offset, the next step is timing recovery and hence the sequence is processed in the timing recovery module. We use a Peak Average Energy Criterion (PAEC) algorithm proposed in [3] to estimate the timing offset $\tau$. If $\tau$ is unknown but deterministic, it has to be estimated in the receiver. We maximize the likelihood function $f(r | \tau)$ with respect to $\tau$, where $r$ is the signal-space vector representation of the received noisy waveform $r(t)$, given by

$$r(t) = s(t; \tau) + n(t)$$

where $r(t)$ is the received signal with an unknown timing offset $\tau$, $s(t; \tau) = \sum_n a_n g(t - nT - \tau)$ is the baseband signal and $\{a_n\}$ are binary $\pm 1$ symbols with equal probability. $n(t)$ is Zero Mean Gaussian Noise. It has been shown mathematically in Ref. [3] that the value of $\tau$ that maximizes the energy is the estimate of timing offset. Hence this is called Peak Average Energy Criterion (PAEC). It is noted that finding $\tau$ that minimizes the energy is more reliable than finding $\tau$ that maximizes the energy. Thus, we will use the technique that finds the value of $\tau_{\text{min}}$, which minimizes the energy and then correcting it to find that $\tau$ which maximizes the energy.

Since we have 12 samples per symbol, the worst case of Inter symbol Interference (ISI) can occur midway between two samples, resulting in maximum timing error (also called timing jitter) of

$$t_{\text{jit}} = \pm \frac{1}{2} \left( \frac{1}{2} \frac{T}{12} \right) = \pm \frac{T}{24}$$

where $T$ is the symbol period.
Fig. 2.5 explains the timing recovery scheme graphically. We maintain Modulo-N (=12) sums of absolute values of the samples (SUM0 to SUM11) under a training sequence of 32 alternate (1, 1) and (0, 0) symbols. These sums are averaged over M (=32) symbol duration. The timing estimation algorithm is given by

\[
\text{sum}(\tau) = \sum_{i=0}^{31} |x(Ni + \tau)|, \text{ where } N = 12 \text{ and } \tau = 0, 1, 2, ..., N - 1
\]

\[
\tau_{\text{min}} = \left\{ \tau : \text{sum}(\tau) = \min_{\tau \in [0, N-1]} \text{sum}(\tau) \right\}
\]

Once the value of \(\tau\) that minimizes the energy, \(\tau_{\text{min}}\), is found, the \(\tau\) corresponding to maximum energy can be calculated as follows:

\[
\tau_{\text{min}} - 6 < 0 \quad (b) \\
\tau_{\text{min}} - 6 > 0
\]
If \((\tau_{\text{min}} - 6) \geq 0\)
then \(\text{Timing Estimate} = \tau_{\text{min}} - 6;\)

\[
\text{elseif } (\tau_{\text{min}} - 6) < 0 \\
\text{then } \text{Timing Estimate} = \tau_{\text{min}} + 6;
\]

It is shown graphically in Fig. 2.6

### 2.2.2 Effect of Frequency Offset Estimate on Timing Recovery

After we have corrected the incoming signal with the estimate of the frequency offset \(\hat{\theta}\), the input to the timing recovery block can be expressed as

\[
r'(k) = \sum_n (a_n + j b_n) g(kT/P - nT + \tau)e^{j(\hat{\theta} - \theta)kT/P} + n(k), \quad \text{where } P = 12
\]

By substituting \(\hat{\theta}' - \hat{\theta} = \varepsilon\), \(r'(k)\) can be expressed as:

\[
r'(k) = \sum_n (a_n + j b_n) g(kT/P - nT + \tau)e^{j\hat{\theta}kT/P} + n(k)
\]

Where the in-phase term is given by

\[
r'(k) = \sum_n (a_n + j b_n) g(kT/P - nT + \tau)e^{j\hat{\theta}kT/P} + n(k)
\]

(2.8)

And the quadrature-phase term is given by

\[
r'(k) = \sum_n (a_n + j b_n) g(kT/P - nT + \tau)e^{j\hat{\theta}kT/P} + n(k)
\]

(2.9)

We see that the in-phase and quadrature components have crosstalk terms \(b_n \sin(\hat{\theta}kT/P)\) and \(a_n \sin(\hat{\theta}kT/P)\) respectively. As \(\varepsilon\) becomes large, the crosstalk terms begin to effect the timing recovery estimation scheme. When \(\varepsilon = 0\), there will be no crosstalk and hence the estimate that we obtain is the best in the presence of noise. We get, at the output of the timing recovery scheme, a number between 0 and 11 (since we have 12 samples per symbol), which indicates the sample closest to the ideal sampling instant. Now, as we increase \(\varepsilon\), then due to the presence of the term \(e^{j\hat{\theta}kT/P}\), the received constellation starts rotating at a rate dependent upon the magnitude of \(\varepsilon\). This is
shown in Fig. 2.7. We see that the incoming symbols trace out a circle as they rotate because of uncompensated frequency offset. A least mean square algorithm based tracking loop is used to track and correct this residual frequency offset continually, which if left uncorrected, can accumulate and cause errors in the decision every time the symbols cross over the decision boundary.

2.3 TRACKING AND SYMBOL DETECTION

Frequency offset estimation and timing recovery is performed only at the start of burst and then continuous tracking and detection are done using the Least Mean Square Algorithm [7].

2.3.1 Least Mean Square Algorithm

The LMS algorithm is a stochastic gradient algorithm. An important feature of the LMS algorithm is its computational simplicity. Consider the
arrangement shown in Fig. 2.8. The error \( e(n) \) is the difference in the filtered output \( y(n) \) and the desired response \( d(n) \)

\[
e(n) = d(n) - y(n)
\]

The purpose of the filter is to produce an estimate of the desired response by adaptively changing the filter coefficients. The filter coefficients are updated using the LMS algorithm with the criterion that the cost function \( J = E\left[|e(n)|^2\right] \) is minimized. The desired response in this case is the constellation points. The error signal \( e(n) \) is the difference between the symbol decisions and slicer input, i.e.,

\[
e_k = [\hat{a}_k + j\hat{b}_k] - [\tilde{a}_k + j\tilde{b}_k]
\]

Here the filter has only one coefficient, which is updated using the standard LMS algorithm adaptation:

\[
w_{k+1} = w_k + 2\mu e_k^* [a'_k + jb'_k]
\]

![Figure 2.8 LMS Structure](image-url)
where $\mu$ is the step size whose value needs to be chosen carefully to ensure rapid convergence of the LMS algorithm. “$w_{k+1}$” is the correction that needs to be applied to the incoming signal.

In summary, algorithms for carrier frequency offset estimation and timing recovery were discussed. Carrier frequency offset estimation algorithm estimates the slope of the best fit line through the unwrapped phase of the received training sequence. Timing recovery algorithm is based upon finding the instant corresponding to the minima of the absolute of the received training sequence and then correcting it to get an estimate of the ideal sampling instant.