CHAPTER 4

ECG COMPRESSION USING NON-LINEAR TRANSFORMS

4.1 Introduction

In this chapter first ENO interpolation techniques which are non-linear in nature are proposed to avoid the discontinuities in the signal. Because of this, the large number of coefficients is not appearing at the edges and this leads to better signal compression as compared to that achieved by using linear transforms.

In this chapter Lifting schemes are proposed for ECG compression which is non-linear wavelet transform techniques. It has been shown in this chapter that LWT based technique performs better at high PRD as compared to the other non-linear transform based techniques.

The main novelties of the research work presented in this chapter are: (i) use of nonlinear transforms for ECG compression, (ii) quality controlled ECG compression and (iii) to further improve the CR, the concept of normalization proposed in chapter 3 is also, analyzed here with nonlinear transform (LWT). In this chapter, the best ways to achieve high CR are presented. In this chapter, it is discussed that which nonlinear transform is best suited for high PRD ECG compression. Quantizer and Encoder/Decoder are explained in chapter 2 and methodology is explained in chapter 3. The chapter represents the non-linear transforms, results and discussions, effect of normalization and concluding remarks.

4.2 Transforms

Transforms are used to obtain a suitable signal representation for efficient source coding. There are two types of transforms: linear and non-linear transforms. The linear transform, decomposes signal into various components by multiplication with a set of transform
functions. Some examples are the Discrete Fourier transform, Discrete cosine transform and Wavelet transform. A linear filtering approach to multiresolution signal decomposition may not be theoretically justified [Heijmans and Goutsias, 2000 (a)]. In particular, the operators used for generating the various levels in a pyramid must crucially depend on the application. The point stressed here is that, scaling an image by means of linear operators may not be compatible with a natural scaling of some image attribute of interest (shape of object). To address this issue, a number of authors have proposed nonlinear multiresolution signal decomposition schemes [Heijmans and Goutsias, 2000 (a)]. Recently Francesc et al. [Arandiga and Donat, 2000] and Sweldens [Heijmans and Goutsias, 2000 (b)] have proposed ENO interpolation and Lifting schemes for image compression respectively. Here, these schemes are used for ECG compression.

4.2.1 Essentially Non-Oscillatory Point-Value (ENOPV) and Cell-Average (ENOCA) Decomposition Techniques

Essentially non-oscillatory point-value (ENOPV) and cell-average (ENOCA) decomposition techniques are a combination of Harten’s multiresolution scheme and ENO interpolation.

To achieve a multiresolution representation data is decomposed into multiple levels. At each level of decomposition, the data is sampled, where by some of the original data points are retained while other are represented by the difference between the original data and a predicted value of that data. The predicted value is obtained by using the data points that are retained. At the reconstruction side, the exact reverse procedure is followed to obtain the reconstructed data. One way of predicting is by interpolation. During the prediction process, interpolation across a discontinuity leads to loss of accuracy and Gibbs phenomenon [Gandhi, 2005]. The ENO interpolatory technique is a data dependent,
nonlinear technique which can eliminate the Gibbs phenomenon [Chan and Zhou, 2001]. In any signal, edges constitute discontinuity in the data. The edges in ECG signal are vital information and it is necessary to preserve them while efficiently representing the signal. Therefore, use of ENO interpolation based technique will be suitable for ECG compression.

4.2.1.1 Harten’s Framework

To understand the concept of Harten’s framework, it can be related to [Gatreuer and Meyer, 2008]. Harten’s framework builds from two operators: Discretization $D_k$ and reconstruction $R_k$. These operators map between functions and discrete signals at resolution $k$ implies finer resolution.

Let $F$ be space of functions, and let $V^k$ be a vector space of discrete signals. Discretization $D_k : F \rightarrow V^k$ is a linear operator mapping $f \in F$ to a discrete signal $v^k = D_k f$.

Reconstruction $R_k : V^k \rightarrow F$ is a (generally) nonlinear operator mapping discrete signals to functions. This nonlinearity is where Harten’s framework is more general than the wavelet framework.

Harten’s framework requires that the discretization and reconstruction operators satisfy a consistency relationship

$$D_k R_k = I_{V^k},$$

(4.1)

Where $I_{V^k}$ denotes identity on $V^k$. The reconstruction $R_k v^k$ must be consistent with the discrete information in $v^k$.

To construct a multi-resolution scheme, define a decimation operator $D^{k-1}_k = D_{k-1} R_k$ and a prediction operator $P^{k}_{k-1} = D_k R_{k-1}$, as follows:
Decimation reduces the discrete signal $v^k$ to $v^{k-1}$; $v^{k-1} = D_{k-1}^k v^k$. The prediction operator predicts $v^k$ from $v^{k-1}$; $P_{k-1}^k v^{k-1}$ is an approximation of $v^k$. Define the predication error as

$$e^k = v^k - P_{k-1}^k v^{k-1}$$

The operators $D_{k-1}^k$ and $P_{k-1}^k$ can be used to construct a multiresolution pyramid. If $v^k$ is the input, then one stage of decomposition outputs the decimated signal $v^{k-1}$ and the prediction error $e^k$. Because $e^k$ is at the same sample rate as $v^k$, the decomposition is 50% oversampled, that is, $(v^{k-1}, e^k)$ redundantly represents $v^k$. This signal decomposition stage is iterated on the decimated signal for a multiresolution representation $(v^{k-L}, e^{k-L+1}, \ldots, e^k)$.

The discretization sequence $Q_{k-1}$ is nested if

$$D_k f = 0 \implies D_{k-1} f = 0 \quad \text{for all} \quad f \in F, k \in \mathbb{Z}. \quad (4.2)$$

If $Q_{k-1}$ is nested, then $D_{k-1}^k$ has no dependence on $R_k$ despite its definition, and it must be a linear operator. This property, along with the consistency requirement (4.1), implies that

$$D_m R_k D_k = D_m \quad \text{for} \quad m \leq k \quad \text{and a discrete analogy of the consistency relationship}$$

$$D_{k-1}^k P_{k-1}^k = D_{k-1} R_k D_k R_{k-1} = D_{k-1} R_{k-1} = I_{v^{k-1}}. \quad (4.3)$$

Consequently, $e^k$ is in the nullspace of $D_{k-1}^k$:

$$D_{k-1}^k e^k = D_{k-1}^k (I_{v^k} - P_{k-1}^k D_{k-1}^k) v^k$$
\[ (D_k^{k-1} - D_k^{k-1} (P_{k-1}^k, D_k^{k-1}))v^k = 0. \] (4.4)

By (4.4), it is possible to design a nonredundant (critically sampled) multiresolution decomposition. Let \( G_k \) be a detail encoder such that \( d^k = G_k e^k \) is at half the sample rate of \( e^k \), and let \( \widetilde{G}_k \) be the corresponding decoder such that \( \widetilde{G}_k G_k e^k = e^k \) for any \( e^k \) in the nullspace of \( D_k^{k-1} \). Then \( \{k-1, d^k\} \) is a nonredundant representation of \( v^k \). This single stage is iterated on the decimated signal for a multiresolution representation \( \{k-L, d^{k-L+1}, \ldots, d^k\} \).

In summary, a multiresolution scheme within Harten’s framework is characterized by six operators: the fundamental discretization and reconstruction operators \( D_k \) and \( R_k \), the decimation operator \( D_k^{k-1} \), the predication operator \( P_{k-1}^k \), and the detail operators \( G_k \) and \( \widetilde{G}_k \).

### 4.2.1.2 ENO Interpolation

Here, ENO interpolation is referred from Gatreuer and Meyer in [Gatreuer and Meyer, 2008] for convenience. The key to the ENO interpolation scheme is the stencil selection [Gandhi, 2005]. ENO interpolation is an effective edge-adaptive strategy for piecewise polynomial interpolation (essentially) without oscillatory artifacts. Let \( v_n = f(x_n) \) be samples from an underlying function \( f \). ENO interpolation approximates \( f \) from the point-values \( v \) by a piecewise polynomial model.

On each subinterval \([x_{n-1}, x_n]\), a polynomial interpolant \( q_n(x) \) is constructed based on a stencil \( S_n \) such that \( q_n(x) = f(x) \) for all \( x \in S_n \). For example, cubic interpolation finds the cubic polynomial \( q_n \) satisfying the point-values at \( S_n = \{x_n-2, x_{n-1}, x_n, x_{n+1}\} \) to interpolate \([x_{n-1}, x_n]\). In ENO interpolation, the \( S_n \) are selected to adapt to the \( v^k \).
The accuracy of such an interpolant depends heavily on the stencil. Let \( S \) be a stencil, and denote by \( \tilde{S} \) its convex hull. If \( S \) has \( M+1 \) points and \( f \in C^{M+1}(\tilde{S}) \), the interpolation error is
\[
|f(x)| - q(x) = \frac{f^{(M+1)}(\xi(x))}{(M+1)!} \prod_{x_j \in S} (x-x_j) \text{ for some } \xi(x) \in \tilde{S}.
\]
(4.5)

If \( f \) is locally smooth, the interpolation error is small. However, if \( f \) has a jump or derivative singularity in \( \tilde{S} \), the error can be much greater.

ENO interpolation attempts to choose stencils that do not cross jumps or derivative discontinuities. To construct an interpolant \( q_n(x) \) of degree \( M \), consider the stencils \( S_n = \{x_{k-1}, \ldots, x_{k+M-1}\} \), \( n - M + 1 \leq k_n \geq n \). Each stencil has \( M+1 \) points and includes \( x_{n-1} \) and \( x_n \). The interpolation error associated with a particular stencil is estimated with the \( M \) th-order divided difference of the stencil samples \( f[S] \):

\[
|f[S]| = \begin{cases} 0, & \text{if } f \text{ is locally polynomial}, \\ \frac{h^{M-p} k^{M-p}}{h^{M-p}}, & \text{if the stencil crosses a discontinuity in } f^{(p)}, \\ \|f^{(p)}\|, & \text{otherwise}, \end{cases}
\]
(4.6)

Where \( \|f^{(p)}\| \) is the size of the jump and \( \|f^{(p)}\| \) is the max-norm of \( f^{(p)} \) over \( \tilde{S} \). For example, if \( M = 2 \), has the form

\[
|f[S]| = \frac{1}{2h^2} |f(x_{k-1}) - 2f(x_k) + f(x_{k+1})|, \quad k = i-1 \text{ or } i.
\]

This estimate distinguishes between intervals where \( f \) is locally linear and intervals containing jumps or first derivative discontinuities. In general, the \( M \) th-order error estimate can detect discontinuities in up to the \( M-1 \) derivative.

In Harten et al. [Harten et al., 1987] consider two methods for choosing the stencil shifts \( k_n \):

Hierarchical stencil selection [Gatreuer and Meyer, 2008]
For each $n$

$$k_n = n$$

for $j = 0, \ldots, M - 2$

if $$|f(x_{k_j+1})| < |f(x_{k_j+1})|,$$

then $k_n := k_n - 1$

end.

Nonhierarchical stencil selection

For each $n$, choose $k_n$ such that

$$|f(x_{k_j+1})| = \min_{n-M+1 \leq k \leq n} |f(x_{k_j+1})|.$$

The hierarchical method has the disadvantage that it can produce singularity crossing for discontinuities in $f'$ or higher derivatives regardless of $M$. The Nonhierarchical method avoid this problem, but it is biased by $f''$. The hierarchical method is usually preferred; however, for the multiresolution constructions, the nonhierarchical method is often the better choice.

Define $I^{ENO}_k : V^k \to F$ as ENO interpolation. For uniformly spaced $x_n$, define

$$(P^{ENO} v)_n = (I^{ENO}_k v)(x_n / 2).$$

$P^{ENO}$ has the property $\downarrow P^{ENO} = I$. An operator with this property called interpolatory operator.

4.2.1.3 Point-Value Discretizations

Consider [Gatreuer and Meyer, 2008] for the ease of clarity of point-value discretizations. ENO interpolation is the reconstruction operator in point-value ENO multiresolution.

Define $D_k$ as sampling point values:

$$v^k_n = (D_k f)_n = f(x^k_n), x^k_n = 2^{-k} n.$$
The interpolant \( R_v v^k = I^{ENO}_k v^k \) is equal to \( f \) at the point-values \( f(x^k_n) \), so the consistency relationship \( D v R = I v \) is trivially satisfied. The decimation operator is downsampling, \( v^{k-1} = D^{k-1} v^k \). The predication operator \( P^{k-1}_{k-1} = D_k R_{k-1} \) is ENO interpolatory prediction, 
\[
(P^{k-1}_{k-1} v^{k-1})_n = (I^{ENO}_k v^{k-1})(x^k_n) = (P^{ENO} v^{k-1})_n.
\]

The prediction error \( e^k = v^k - P^{k-1}_{k-1} v^{k-1} \) is nonzero only for odd \( n \). For a nonredundant representation, the detail is encoded by keeping samples at odd \( n \); \( d^k_n = e^k_{2n+1} \).

**4.2.1.4 Cell-Average Discretizations**

Gatreuer and Meyer in [Gatreuer and Meyer, 2008] is referred to clear the concept of cell-average discretizations. In most multiresolution schemes, smoothing is applied before downsampling to avoid aliasing in the coarser subbands. The cell-average discretizations use local averages to achieve this smoothing. Suppose \( f \in L^1_{loc} \) is discretized by computing weighted averages
\[
\widehat{Q}_k f = 2^k \int 2^k \phi(n-2^k x) f(x) dx,
\]
where \( \phi \) is a compactly supported weighting function. Equivalently,
\[
\widehat{Q}_k f = \widehat{Q}_k \ast f = \int \phi \delta \text{, with } \phi(x) = 2^k \phi(2^k x). \]
For cell-average discretization, \( \phi \) is the Haar scaling function
\[
\phi(x) = \begin{cases} 
1, & 0 \leq x < 1, \\
0, & \text{otherwise}
\end{cases}
\]

The discretization \( \widehat{Q}_k f \) is the average over the cell \( c^k_n = (2^{-k} (n-1), 2^{-k} n) \),
\[
\widehat{Q}_k f = 2^k \int_{c^k_n} f(x) dx,
\]
hence the name “cell-average.” The dilation equation of the Haar scaling function \( \phi(x) = \phi(2x) + \phi(2x-1) \) implies that \( v_n^{k-1} = \frac{1}{2} v_{2n}^k + \frac{1}{2} v_{2n-1}^k \). Therefore, the decimation operator is 
\[
(D_{k}^{k-1}v^k)_{n} = \frac{1}{2} v_{2n}^k + \frac{1}{2} v_{2n-1}^k.
\]

As with cell-average discretization, the dilation equation of \( \phi \) determines \( D_{k}^{k-1} \):
\[
\phi(x) = \frac{1}{2} \phi(2x-1) + \phi(2x) + \frac{1}{2} \phi(2x+1) ,
\]
\[
(D_{k}^{k-1}v^k)_{n} = \frac{1}{4} v_{2n}^k + \frac{1}{2} v_{2n}^k + \frac{1}{4} v_{2n+1}^k
\]

Consider the reconstruction operators for these discretization operators. To satisfy the consistency requirement \( D_{k} R_{k} = I_{\psi} \), one approach is to modify the point-value ENO interpolation such that the reconstruction function attains averages rather than point-values.

Harten’s approach to constructing \( R_{k} \) is reconstruction via primitive function. The idea is to reduce reconstruction to interpolation (that is, to point-value reconstruction). For cell-average discretization, the relationship between \( f \) and its primitive \( \hat{f} \) is
\[
\hat{f}(x) = \int_{0}^{x} f(y)dy, \quad f(x) = \frac{d}{dx} \hat{f}(x)
\]

Set \( \dot{v}_{\triangle}^k = 0 \), then the relationship between \( v^k \) and \( \dot{v}^k \) is
\[
\dot{v}^k_n = 2^{-k} \sum_{m=0}^{n} v^k_m, \quad v^k_n = 2^{k} (v^k_n - \dot{v}^k_{n-1}) , \quad n = 0, \ldots, J_{k} - 1. \tag{4.7}
\]

Let \( I_{k} \) be any interpolatory operator, for example, ENO interpolation. Define the reconstruction operator as
\[
(R_{k}v^k)(x) = \frac{d}{dx} I_{k} \dot{v}^k (x) \tag{4.8}
\]
This $R_k$ satisfies the consistency relationship: For any $v^k \in V^k$,

$$(D_k R_k v^k)_n = 2^k \int_{x_{n-1}}^{x_n} \frac{d}{dx} (I_k \dot{v}^k) dx = 2^k (\dot{v}_n^k - \dot{v}_{n-1}^k) = v_n^k$$

The derivative in (4.8) should be interpreted in the weak sense. When $I_k$ is ENO interpolation, $I_k \dot{v}^k$ is piecewise differentiable, so $R_k \dot{v}^k$ is piecewise continuous.

Using (4.7) to obtain $\dot{v}^k$ from $v^k$, the prediction operator $P_{k-1}^k = D_k R_{k-1}$ is

$$(P_{k-1}^k v^{k-1})_n = 2^k [(I_{k-1} \dot{v}^{k-1})(x_n^k) - (I_{k-1} \dot{v}^{k-1})(x_{n-1}^k)]$$

(4.9)

The relationship between $f$ and its primitive is

$$\hat{f}(x) = \int_0^x f(z)dzdy, \quad f(x) = \frac{d^2}{dx^2} \hat{f}(x)$$

Setting $\dot{v}_n^k = 0$ for $n < 0$, there is a bijection between $v^k$ and $\dot{v}^k$:

$$\dot{v}^k = 4^k \sum_{m=0}^n \sum_{j=0}^m v_j^k, \quad v_n^k = 4^k (\dot{v}_{n+1}^k - 2\dot{v}_n^k + \dot{v}_{n-1}^k), \quad n = 0, ..., J_k - 1$$

Define the reconstruction operator analogously to (4.8),

$$(R_k v^k)(x) = \frac{d^2}{dx^2} I_k \dot{v}^k(x)$$

(4.10)

The second derivative should be interpreted in the weak sense. While the first derivative in (4.8) can produce jump discontinuities, (4.10) can produce Dirac measure.

It is true that (4.10) satisfies $D_k R_k = I_{v^k}$, but verifying this is more cumbersome that in the cell-average case. The prediction operator $P_{k-1}^k = D_k R_{k-1}$ is

$$(P_{k-1}^k v^{k-1})_n = 4^k [(I_{k-1} \dot{v}^{k-1})(x_{n+1}) - 2(I_{k-1} \dot{v}^{k-1})(x_n) + (I_{k-1} \dot{v}^{k-1})(x_{n-1})]$$

(4.11)

Detail encoder and decoder operators can be found using the property that $D_k e^k = 0$. For cell-average discretization, this is $e^k_{2n} = -e^k_{2n-1}$; thus a choice of detail encoder and decoder operators is
\[ d_n^k = (G_k e^k)_n = e_{2n-1}^k, \quad \begin{cases} e_{2n-1}^k = \left( \tilde{G}_k d^k \right)_{2n-1} = d_n^k, \vspace{1em} \\ e_{2n}^k = \left( \tilde{G}_k d^k \right)_{2n} = -d_n^k. \end{cases} \]

Cell-average discretization is member of a family of spline-based discretization. Denote the order of a spline discretization by \( N \), with cell-average discretization as \( N = 1 \) and point-value discretization fits into this classification as \( N = 0 \).

For general \( N \), the weight function \( \phi \) is the B-spline of order \( N - 1 \), and \( \hat{f} \) is related to \( f \) through a \( N \) th-order integral.

**4.2.2 The Lifting Scheme**

Lifting amounts to modifying the analysis and synthesis operators in such a way that the properties of the modified scheme are “better” than those of original one. Here “better” can be interpreted in different ways. For example, in the linear case, it may mean that the number of vanishing moments is larger. Lifting can be used to construct wavelet decompositions for signals that are defined on arbitrary domains, or to construct nonlinear coupled or uncoupled wavelet decomposition. Lifting schemes has many advantages over the WT which is traditionally implemented by convolution based approach or FIR filter bank structures. These are discussed as follows (i) it requires less computation (upto 50%) compared to the convolution based approach (ii) during lifting implementation, no extra buffer is required because of the in-place computation of lifting (iii) it offers integer to integer transformation suitable for lossless compression [Acharya and Chakrabarti, 2006].

The lifting scheme consists of two steps: a prediction step followed by an update step [Heijmans and Goutsias, 2000 (b)].

**4.2.2.1 Prediction Lifting**

For better understanding of prediction lifting, [Heijmans and Goutsias, 2000 (b)] is described. Consider one level of a given couples wavelet decomposition with analysis operators \( \psi^\dagger : V_0 \rightarrow V_1 \), \( \omega^\dagger : V_0 \rightarrow W_1 \) and \( \Psi^\dagger : V_1 \times W_1 \rightarrow V_0 \). In many applications, such as
compression, the goal is to find wavelet schemes that produce small detail signals. Starting
from a given scheme, to decrease the detail signal \( y_i \) by utilizing signal information
contained in the approximation signal \( x_i = \psi^\dagger \zeta_{i,0} \). Therefore use a prediction operator
\( \pi : V_i \rightarrow W_i \), and put
\[
y_i = y_i - \pi \zeta_{i,1}
\]
(4.12)
It is evident that the original signal can be reconstructed from the data in \( x_i, y_i \), since
\[
x_0 = \Psi^\dagger \zeta_{i,1}, y_i = \Psi^\dagger \zeta_{i,1}, y_i + \pi \zeta_{i,1}
\]
(4.13)
Thus the prediction lifted scheme:
\[
\psi_p^\dagger (x) = \psi^\dagger (x), \quad x \in V_0
\]
(4.14)
\[
\omega_p^\dagger (x) = \omega^\dagger (x) - \pi \psi^\dagger (x), \quad x \in V_0
\]
(4.15)
\[
\Psi_p^\dagger (x, y) = \Psi^\dagger (x, y + \pi (x)) \quad x \in V_i, y \in W_i
\]
(4.16)

4.2.2.2 Update Lifting

Heijmans and Goutsias in [Heijmans and Goutsias, 2000 (b)] is consulted, so as to easily
understand the concept of update lifting. Analogously, choose to modify the
approximation signal \( x_i \) using the information in \( y_i \). To get an updated approximation
signal by putting
\[
x_i = x_i - \lambda (y_i)
\]
(4.17)
Here \( \lambda \) is an operator mapping \( W_i \) into \( V_i \) called the update lifting operator. In practice we
will choose \( \lambda \) in such a way that the resulting analysis operator \( x_0 \mapsto x_i \) satisfies a certain
constraint. In the linear case it is often required that this operator is a low-pass filter.
Alternatively, we may impose the condition that it preserves a given signal attribute (e.g.,
average or maximum). If the operator \( x_0 \mapsto x_i \) does not satisfy the constraint, we may choose \( \lambda \) in such a way that \( x_0 \mapsto x_i \), with \( x_i \) given by (4.17), does satisfy the constraint.

The input signal \( x_0 \) can be reconstructed from the data in \( x_i, y_i \), since

\[
x_0 = \Psi^+(x_i, y_i) = \Psi^+(x_i + \lambda(y_i), y_i).
\]

Thus we are led to the update-lifted scheme

\[
\psi^+(x) = \psi^+(x) - \lambda \omega^+(x), \quad x \in V_0
\]

(4.19)

\[
\omega^+(x) = \omega^+(x), \quad x \in V_0
\]

(4.20)

\[
\Psi^+(x, y) = \Psi^+(x + \lambda(y), y), \quad x \in V_i, y \in W_i
\]

(4.21)

### 4.2.2.3 Maxlift Transform

To describe the computational scheme of Maxlift transform Heijmans and Goutsias in [Heijmans and Goutsias, 2000 (b)] is taken. It starts from the so-called lazy wavelet decomposition, which is nothing but a separation of the input signal \( x_0 \) into odd and even samples:

\[
x_i(n) = \psi^+(x_0)(n) = x_0(2n), \quad y_i(n) = \omega^+(x_0)(n) = x_0(2n + 1)
\]

We use the maximum of \( x_i(n) \) and \( x_i(n + 1) \) as a prediction for \( y_i(n) \), i.e.

**prediction:** \( y_i(n) = y_i(n) - (x_i(n) \vee x_i(n + 1)) \).

For the update step

**update:** \( x_i(n) = x_i(n) + (0 \vee y_i(n - 1) \vee y_i(n)) \)

The update step is chosen in such a way that local maxima of the input signal \( x_0 \) are mapped to the approximation signal \( x_i \).

Writing \( x = x_i \) and \( y = y_i \), the following properties can be established:
1. Local maxima of $x_0$ at even points are mapped to the approximation signal. More precisely:

$$x_0(2n) \leq x(n) \leq x_0(2n - 1) \lor x_0(2n) \lor x_0(2n + 1).$$

2. Local maxima of $x_0$ at odd points in a neighbourhood of five points are mapped to the approximation signal: if $x_0(2n + 1) \geq x_0(2n + 1 + i)$ for $i = -2, -1, 0, 1, 2$, then

$$x_0(2n + 1) = \begin{cases} x(n), & x_0(2n + 2) \leq x_0(2n) \\ x(n + 1), & x_0(2n + 2) \geq x_0(2n) \end{cases}$$

3. Furthermore, we can show that $x_0(2n + 1) \leq x(n) \lor x(n + 1)$.

4. No new maxima are created: if the approximation signal $x$ has a local maximum at $n$, that is, $x(n - 1), x(n + 1) \leq x(n)$, then, for some $m \in \mathbb{Z} - 1, 2n, 2n + 1$, the original signal $x_0$ has a local maximum at $m$ and $x_0(m) = x(n)$.

Our computation consists of three steps:

1. Analysis $x_0 \mapsto x_4, y_4, y_3, y_2, y_1$;

2. Filtering $y_j \mapsto \tilde{y}_j = f(y_j)$, and

3. Synthesis $x_4, \tilde{y}_4, \tilde{y}_3, \tilde{y}_2, \tilde{y}_1 \mapsto \tilde{x}_0$.

### 4.2.2.4 Medlift Transform

To clear the concept of Medlift transform, Heijmans and Goutsias in [Heijmans and Goutsias, 2000 (b)] is considered. Let take $\sim, -$ to be standard subtraction, and $\hat{+}, \hat{+}, \hat{+}$ to be standard addition. Consider the case of a prediction-update lifting scheme with initial signal decomposition given by means of the lazy wavelet, and prediction and update operators given by

$$\pi(x)(n) = x(n), \quad \lambda(y)(n) = \text{median}(0, y(n - 1), y(n))$$ \quad \text{(4.22)}
Obtain an uncoupled wavelet decomposition scheme, with analysis and synthesis operators given by

\[ \psi_{pu}^\wedge (x)(n) = x(2n) + \text{median}(0, x(2n-1) - x(2n-2), x(2n+1) - x(2n)) \]  

(4.23)

\[ \omega_{pu}^\wedge (x)(n) = x(2n+1) - x(2n) \]  

(4.24)

\[ \psi_{pu}^\dagger (x)(2n) = \psi_{pu}^\wedge (x)(2n+1) = x(n) \]  

(4.25)

\[ \omega_{pu}^\dagger (y)(2n) = -\text{median}(0, y(n-1), y(n)) \]  

(4.26)

Notice that, the update operator adjusts the value of \( x(2n) \) based on the local structure of the input signal \( x(n) \). If the difference \( x(2n-1) - x(2n-2) \) is negative (or positive) and the difference \( x(2n+1) - x(2n) \) is positive (or negative), then no adjustment is made. This happens, for example, when \( x(2n) \) is a local minimum (or maximum). If however both differences \( x(2n-1) - x(2n-2) \) and \( x(2n+1) - x(2n) \) are negative (or positive), then \( x(2n) \) is adjusted by adding the smallest (in absolute value) difference. For example, when \( x(n) \) (locally) oscillates between two values, then (4.23) will bring in line with \( x(2n-1) \), thus getting a scaled signal \( \psi_{pu}^\wedge (x) \) that approximates \( x \) “better” than the scaled signal \( \psi^\wedge (x) \) before prediction-update lifting. Concerning the last property, one may observe that it holds for positive as well as for negative constants \( c \).

Alternatively, we may choose

\[ \pi(x)(n) = \frac{1}{2} (x(n) + x(n+1)) \]

and \( \lambda(y) \) as in (4.22). This choice leads to an uncoupled wavelet decomposition scheme that has two “vanishing moments,” in the sense that the detail signal, resulting from an input signal \( x(n) = an + b \), will be zero.
Finally, one can replace the previous linear prediction operator, with the nonlinear prediction operator

\[ \pi(x)(n) = \text{median}(x(n-1), x(n), x(n+1)). \]

This choice, together with (4.22) for the update operator, leads to a coupled wavelet decomposition scheme.

### 4.2.2.5 Lifting Wavelet Transform (LWT)

LWT is similar to DWT except that the numbers of samples at each stage is same as the initial set of samples. The input samples are split into odd or even sets of samples and passed through the filters (lifting steps) to give rise to approximation and details. Since the number of samples to be stored is same as that of the input at each stage, which save memory. The number of computations required is also reduced, since the approximation coefficients at one level can be derived from the detailed coefficients already computed and some of the input samples. The integer wavelet coefficients are also possible with perfect reconstruction. LWT provides ease in implementation in hardware [Ramesh and Ranjit, 2002].

The main features of the Lifting based DWT scheme is to break up the highpass and lowpass filters into a sequence of upper and lower triangular matrices and convert the filter implementation into banded matrix multiplications. Such scheme has several advantages, including “in-place” computation of the DWT, integer-to-integer wavelet transform (IWT), symmetric forward and inverse transform, etc [Andra et al., 2002].

The basic principle of the lifting scheme is to factorize the polyphase matrix of a wavelet filter into a sequence of alternating upper and lower triangular matrices and a diagonal matrix [Andra et al., 2002].
Let $\tilde{h}(z)$ and $\tilde{g}(z)$ be the lowpass and highpass analysis filters, and let $h(z)$ and $g(z)$ be the lowpass and highpass synthesis filters. The corresponding polyphase matrices are defined as Acharya and Chakrabarti in [Acharya and Chakrabarti, 2006]

$$\tilde{P}(z) = \begin{bmatrix} \tilde{h}_e(z) & \tilde{h}_o(z) \\ \tilde{g}_e(z) & \tilde{g}_o(z) \end{bmatrix} \quad \text{and} \quad P(z) = \begin{bmatrix} h_e(z) & h_o(z) \\ g_e(z) & g_o(z) \end{bmatrix}.$$ 

If $\tilde{h}, \tilde{g}$ is a complementary filter pair, then $\tilde{P}(z)$ can always be factored into lifting steps as

$$\tilde{P}_1(z) = \begin{bmatrix} K & 0 \\ 0 & \frac{1}{K} \end{bmatrix} \prod_{i=1}^{m} \begin{bmatrix} 1 & \tilde{s}_i(z) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ \tilde{t}_i(z) & 1 \end{bmatrix}$$

or

$$\tilde{P}_2(z) = \begin{bmatrix} K & 0 \\ 0 & \frac{1}{K} \end{bmatrix} \prod_{i=1}^{m} \begin{bmatrix} 1 & \tilde{t}_i(z) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ \tilde{s}_i(z) & 1 \end{bmatrix}$$

Where $K$ is a constant. There are two types of lifting schemes.

Scheme 1, which corresponds to the $\tilde{P}_1(z)$ factorization, consists of three steps:

1) Predict step, where the even samples are multiplied by the time domain equivalent of $\tilde{t}(z)$ and are added to the odd samples;

2) Update step, where updated odd samples are multiplied by the time domain equivalent of $\tilde{s}(z)$ and are added to the even samples;

3) Scaling step, where the even samples are multiplied by $1/K$ and odd samples by $K$.

The inverse DWT is obtained by traversing in the reverse direction, changing the factor $K$ to $1/K$, factor $1/K$ to $K$, and reversing the signs of coefficients in $\tilde{t}(z)$ and $\tilde{s}(z)$ [Andra et al., 2002].

In Scheme 2, which corresponds to the $\tilde{P}_2(z)$ factorization, the odd samples are calculated in the first step, and the even samples are calculated in the second step. The inverse is obtained by traversing in the reverse direction [Andra et al., 2002].
Due to the linearity of the lifting scheme, if the input data is in integer format, it is possible to maintain data to be in integer format throughout the transform by introducing a rounding function in the filtering operation. Due to this property, the transform is reversible (i.e., lossless) and is called the integer wavelet transform (IWT). It should be noted that filter coefficients need not be integers for IWT. However, if a scaling step is present in the factorization, IWT cannot be achieved [Andra et al., 2002].

4.3 Result and Discussions

The performance of different non-linear transforms (as reported in section 4.2) is tested by experimentation on the well known ECG database, MIT-BIH Arrhythmia. Out of the tested transforms, the transforms namely ENOPV, ENOCA, Maxlift, and Medlift has never been used for ECG compression till date. Each record contains 11 bit resolution and sampling frequency of 360 Hz. The ECG signal is transformed using ENOPV with 3 stages decomposition with cubic interpolation. The ECG signal is transformed using ENOCA with three stages and five degree interpolation. In case of Maxlift and Medlift transformation levels used for decomposition are 4 and convergence precision $\varepsilon$ is 3%. In LWT, ECG signal is transformed using cdf 9/7 and decomposed to 5 levels.

The results are presented in Tables 4.1-4.4, represents the CR at fixed PRD=1, PRD=2 and PRD=3 respectively for different ECG signals. Table 4.5 represent the CR at fixed PRD=0.5, PRD=1, PRD=2 and PRD=3 respectively for different ECG signal. From the numerical results, it is observed that PRD before quantization (BPRD) is nearly equal to PRD after quantization (QPRD). The results are presented on different ECG signals of varying characteristics.

The limitation of ENOPV is that the samples of the signal must be the power of 2 that is why in Table 4.1 the samples taken are 65536 (3.0341 min duration) and for the other ENOCA, Maxlift, Medlift, and LWT (Table 4.2- Table 4.5) there is no such limitation so
the number of samples taken are 43200 (2 min duration). In Table 4.2, the non zero coefficients after thresholding are quantized with 14 bit Max-Lloyd quantizer and for the others Table 4.1 and Table 4.3- Table 4.5 the non zero coefficients after thresholding are quantized with 12 bit Max-Lloyd quantizer. Table 4.6 compares the performance of different non-linear transforms used in this thesis with that reported in the literature. It shows that the performance of ENOPV, ENOCA and LWT (38.75, 51.42 and 56.14) give high CR in comparison to previous work done.

Next, to show the performance of non-linear transforms on different ECG signals (number of samples are 65536, Time=3.0341 min) at different PRDs (ranging from 0.5 to 3), results are presented graphically in Figure 4.1 to Figure 4.3 and compared to the linear transform (DCT and WT). Chapter 3 concludes that DCT and WT is best suited from linear transforms for ECG compression. Here, the non zero coefficients after thresholding are quantized with 12 bit Max-Lloyd quantizer.

Figure 4.1 shows that for record MIT-BIH 117 the performance behavior of Maxlift and Medlift is nearly same for the total experimental range that is 0.5 to 3 PRD, CR just increases from 10.41 and 10.42 to 16.65 and 16.84. In case of ENOPV and ENOCA, CR increases from 11.18 and 10.54 to 38.75 and 39.29 almost linearly for the PRD range from 0.5 to 2 then for 2 to 3 it increases 53.16 and 53.32. The performance behavior of LWT is approximately linear with positive slope and for the total experimental range that is 0.5 to 3 CR increases from 11.99 to 85.57.

The performance behavior of ENOPV, ENOCA and LWT is better than DCT and WT (linear transforms) at high (3) PRD and at low (0.5) PRD DCT gives better performance than nonlinear transforms.

From these figures it can be concluded that the performance behavior of non-linear transforms is same for low PRD=0.5 and for high PRD=3, maximum CR is achieved with
LWT. Hence LWT more suitable as compared with non-linear transforms. For low (0.5) PRD the linear transform DCT is better than among linear and nonlinear transforms. Similarly observation can be made from the Figure 4.2 to Figure 4.3.

Further, for visual comparison of ECG compression using non-linear transforms tested at low PRD (1) and high PRD (3) on normal (MIT-BIH 117) rhythm and abnormal (MIT-BIH 232) rhythm. The original and reconstructed signal along with error signal is shown in Figure 4.4- Figure 4.23. The closer look on figures reveals that reconstructed signal is almost identical to original signal.

**Table 4.1: Performance of ENOPV transform on different ECG signals**

<table>
<thead>
<tr>
<th>Signal</th>
<th>(^\text{2UPRD}=1)</th>
<th>(^\text{2UPRD}=2)</th>
<th>(^\text{2UPRD}=3)</th>
<th>(^\text{3BPRD})</th>
<th>(^\text{3QPRD})</th>
<th>(^\text{3CR})</th>
<th>(^\text{4BPRD})</th>
<th>(^\text{4QPRD})</th>
<th>(^\text{4CR})</th>
<th>(^\text{5BPRD})</th>
<th>(^\text{5QPRD})</th>
<th>(^\text{5CR})</th>
</tr>
</thead>
<tbody>
<tr>
<td>121</td>
<td>1.00</td>
<td>1.01</td>
<td>21.52</td>
<td>1.99</td>
<td>1.99</td>
<td>59.95</td>
<td>3.00</td>
<td>3.00</td>
<td>74.84</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>122</td>
<td>1.00</td>
<td>1.02</td>
<td>13.85</td>
<td>1.98</td>
<td>1.98</td>
<td>28.07</td>
<td>3.01</td>
<td>3.01</td>
<td>42.54</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>205</td>
<td>1.00</td>
<td>1.03</td>
<td>11.76</td>
<td>1.95</td>
<td>2.14</td>
<td>20.27</td>
<td>3.00</td>
<td>3.00</td>
<td>28.08</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>103</td>
<td>0.99</td>
<td>1.25</td>
<td>11.23</td>
<td>1.97</td>
<td>2.10</td>
<td>15.26</td>
<td>3.02</td>
<td>3.02</td>
<td>25.74</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>123</td>
<td>0.99</td>
<td>1.00</td>
<td>15.93</td>
<td>2.01</td>
<td>2.01</td>
<td>33.90</td>
<td>3.02</td>
<td>3.02</td>
<td>48.36</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>116</td>
<td>1.00</td>
<td>1.01</td>
<td>18.02</td>
<td>1.99</td>
<td>2.01</td>
<td>29.28</td>
<td>2.96</td>
<td>2.98</td>
<td>34.73</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>115</td>
<td>0.99</td>
<td>1.00</td>
<td>15.11</td>
<td>1.99</td>
<td>1.99</td>
<td>34.49</td>
<td>3.02</td>
<td>3.02</td>
<td>47.12</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>233</td>
<td>0.99</td>
<td>1.03</td>
<td>12.69</td>
<td>1.99</td>
<td>2.01</td>
<td>20.67</td>
<td>2.99</td>
<td>3.00</td>
<td>25.36</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>109</td>
<td>0.99</td>
<td>1.11</td>
<td>11.52</td>
<td>2.00</td>
<td>2.01</td>
<td>18.96</td>
<td>2.97</td>
<td>2.98</td>
<td>36.14</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>221</td>
<td>0.99</td>
<td>1.13</td>
<td>11.30</td>
<td>2.01</td>
<td>2.05</td>
<td>11.88</td>
<td>3.02</td>
<td>3.03</td>
<td>15.19</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

1\(^\text{Qbits- bits used for quantization}\) 2\(^\text{UPRD- user defined PRD}\)
3\(^\text{BPRD- PRD before quantization}\) 4\(^\text{QPRD- PRD after quantization}\) 5\(^\text{CR- Compression ratio}\)
Table 4.2: Performance of ENOCA transform on different ECG signals

<table>
<thead>
<tr>
<th>Signal</th>
<th>^2UPRD=1</th>
<th>^2UPRD=2</th>
<th>^2UPRD=3</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>^BPRD</td>
<td>^QPRD</td>
<td>^CR</td>
</tr>
<tr>
<td>121</td>
<td>1.00</td>
<td>1.00</td>
<td>11.78</td>
</tr>
<tr>
<td>122</td>
<td>0.99</td>
<td>1.01</td>
<td>10.70</td>
</tr>
<tr>
<td>223</td>
<td>1.00</td>
<td>1.00</td>
<td>10.48</td>
</tr>
<tr>
<td>220</td>
<td>0.99</td>
<td>1.01</td>
<td>10.02</td>
</tr>
<tr>
<td>119</td>
<td>1.00</td>
<td>1.00</td>
<td>10.97</td>
</tr>
<tr>
<td>123</td>
<td>0.99</td>
<td>0.99</td>
<td>10.61</td>
</tr>
<tr>
<td>116</td>
<td>1.00</td>
<td>1.00</td>
<td>11.34</td>
</tr>
<tr>
<td>115</td>
<td>0.99</td>
<td>0.99</td>
<td>11.10</td>
</tr>
<tr>
<td>233</td>
<td>0.99</td>
<td>1.00</td>
<td>11.47</td>
</tr>
</tbody>
</table>

1 Qbits - bits used for quantization
2 UPRD - user defined PRD
3 BPRD - PRD before quantization
4 QPRD - PRD after quantization
5 CR - Compression ratio
Table 4.3: Performance of Maxlift transform on different ECG signals

Maxlift Transform, \(^1\)Qbits=14, Samples=43200, Time=2 min, Decomposition Level=4, \(\varepsilon=3\%\)

<table>
<thead>
<tr>
<th>Signal</th>
<th>(^1)UPRD=1</th>
<th>(^2)UPRD=2</th>
<th>(^3)UPRD=3</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(^1)BPRD</td>
<td>(^1)QPRD</td>
<td>(^1)CR</td>
</tr>
<tr>
<td>121</td>
<td>1.27</td>
<td>1.27</td>
<td>11.28</td>
</tr>
<tr>
<td>122</td>
<td>1.04</td>
<td>1.05</td>
<td>9.61</td>
</tr>
<tr>
<td>205</td>
<td>1.24</td>
<td>1.51</td>
<td>10.51</td>
</tr>
<tr>
<td>123</td>
<td>1.16</td>
<td>1.16</td>
<td>9.68</td>
</tr>
<tr>
<td>116</td>
<td>1.09</td>
<td>1.12</td>
<td>9.03</td>
</tr>
<tr>
<td>115</td>
<td>0.91</td>
<td>0.91</td>
<td>9.39</td>
</tr>
<tr>
<td>223</td>
<td>1.38</td>
<td>1.38</td>
<td>9.93</td>
</tr>
<tr>
<td>220</td>
<td>1.38</td>
<td>1.38</td>
<td>9.22</td>
</tr>
<tr>
<td>119</td>
<td>1.00</td>
<td>1.00</td>
<td>9.63</td>
</tr>
<tr>
<td>118</td>
<td>0.89</td>
<td>0.89</td>
<td>9.53</td>
</tr>
</tbody>
</table>

\(^1\)Qbits- bits used for quantization  \(^2\)UPRD- user defined PRD

\(^3\)BPRD- PRD before quantization  \(^4\)QPRD- PRD after quantization  \(^5\)CR- Compression ratio
Table 4.4: Performance of Medlift transform on different ECG signals

<table>
<thead>
<tr>
<th>Signal</th>
<th>(^2)UPRD=1</th>
<th>(^2)UPRD=2</th>
<th>(^2)UPRD=3</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(^3)BPRD</td>
<td>(^4)QPRD</td>
<td>(^5)CR</td>
</tr>
<tr>
<td>121</td>
<td>1.20</td>
<td>1.21</td>
<td>11.68</td>
</tr>
<tr>
<td>122</td>
<td>0.98</td>
<td>0.98</td>
<td>10.04</td>
</tr>
<tr>
<td>205</td>
<td>1.22</td>
<td>1.22</td>
<td>11.06</td>
</tr>
<tr>
<td>123</td>
<td>1.07</td>
<td>1.08</td>
<td>9.73</td>
</tr>
<tr>
<td>116</td>
<td>1.17</td>
<td>1.17</td>
<td>9.10</td>
</tr>
<tr>
<td>115</td>
<td>0.89</td>
<td>0.89</td>
<td>9.51</td>
</tr>
<tr>
<td>223</td>
<td>1.33</td>
<td>1.33</td>
<td>10.05</td>
</tr>
<tr>
<td>220</td>
<td>1.33</td>
<td>1.33</td>
<td>9.38</td>
</tr>
<tr>
<td>119</td>
<td>1.23</td>
<td>1.23</td>
<td>10.01</td>
</tr>
<tr>
<td>118</td>
<td>0.91</td>
<td>0.91</td>
<td>9.70</td>
</tr>
</tbody>
</table>

\(^1\)Qbits- bits used for quantization  \(^2\)UPRD- user defined PRD
\(^3\)BPRD- PRD before quantization  \(^4\)QPRD- PRD after quantization  \(^5\)CR-Compression ratio
Table 4.5: Performance of LWT on different ECG signals

LWT, \(^1\)Qbits=12, Samples=43200, Time=2 min,

<table>
<thead>
<tr>
<th>Signal</th>
<th>(^2)UPRD=0.5</th>
<th>(^3)UPRD=1</th>
<th>(^4)UPRD=2</th>
<th>(^5)UPRD=3</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(^6)BPRD</td>
<td>(^7)QPRD</td>
<td>(^8)CR</td>
<td>(^9)BPRD</td>
</tr>
<tr>
<td>121</td>
<td>0.49</td>
<td>0.50</td>
<td>13.17</td>
<td>1.00</td>
</tr>
<tr>
<td>122</td>
<td>0.49</td>
<td>0.49</td>
<td>11.39</td>
<td>1.00</td>
</tr>
<tr>
<td>205</td>
<td>0.49</td>
<td>0.49</td>
<td>11.56</td>
<td>1.00</td>
</tr>
<tr>
<td>103</td>
<td>0.50</td>
<td>0.52</td>
<td>11.03</td>
<td>0.99</td>
</tr>
<tr>
<td>123</td>
<td>0.49</td>
<td>0.49</td>
<td>11.76</td>
<td>1.00</td>
</tr>
<tr>
<td>116</td>
<td>0.50</td>
<td>0.50</td>
<td>12.47</td>
<td>0.99</td>
</tr>
<tr>
<td>219</td>
<td>0.49</td>
<td>0.49</td>
<td>12.42</td>
<td>1.00</td>
</tr>
<tr>
<td>115</td>
<td>0.50</td>
<td>0.51</td>
<td>11.23</td>
<td>1.00</td>
</tr>
<tr>
<td>223</td>
<td>0.49</td>
<td>0.50</td>
<td>11.63</td>
<td>0.99</td>
</tr>
<tr>
<td>233</td>
<td>0.50</td>
<td>0.51</td>
<td>11.67</td>
<td>1.00</td>
</tr>
</tbody>
</table>

\(^1\)Qbits- bits used for quantization   \(^2\)UPRD- user defined PRD
\(^3\)BPRD- PRD before quantization   \(^4\)QPRD- PRD after quantization   \(^5\)CR- Compression ratio
Table 4.6: Comparison of different non-linear transforms with literature

<table>
<thead>
<tr>
<th>Transform</th>
<th>PRD</th>
<th>CR</th>
</tr>
</thead>
<tbody>
<tr>
<td>ENOPV[Aggarwal and Patterh, 2012b]</td>
<td>2.00</td>
<td>38.75</td>
</tr>
<tr>
<td>ENOCA[Aggarwal and Patterh, 2009a]</td>
<td>2.69</td>
<td>51.42</td>
</tr>
<tr>
<td>MAXLIFT[Aggarwal and Patterh, 2009c]</td>
<td>2.66</td>
<td>16.58</td>
</tr>
<tr>
<td>MEDLIFT[Aggarwal and Patterh, 2009c]</td>
<td>2.67</td>
<td>17.07</td>
</tr>
<tr>
<td>LWT[Aggarwal and Patterh, 2013]</td>
<td>1.99</td>
<td>56.14</td>
</tr>
<tr>
<td>ECG compression with retrieved quality guaranteed. [Velasco et al., 2004]</td>
<td>2.5359</td>
<td>17.40</td>
</tr>
<tr>
<td>Fixed percentage of wavelet coefficients to be zeroed for ECG compression. [Benzid et al., 2003]</td>
<td>2.5518</td>
<td>16.24</td>
</tr>
<tr>
<td>Electrocardiogram compression method based on the adaptive wavelet coefficients quantization combined to a modified two-role encoder. [Benzid et al., 2007]</td>
<td>2.15</td>
<td>16.70</td>
</tr>
<tr>
<td>Constrained ECG compression algorithm using block-based discrete cosine transfrom. [Benzid et al., 2008]</td>
<td>2.54</td>
<td>21.74</td>
</tr>
<tr>
<td>Wavelet compression of ECG signals using SPIHT algorithm. [Pooyan et al., 2004]</td>
<td>3.1</td>
<td>21.4</td>
</tr>
</tbody>
</table>

Figure 4.1: Comparison of non-linear different transforms on record, MIT-BIH 117
Figure 4.2: Comparison of non-linear different transforms on record, MIT-BIH 112

Figure 4.3: Comparison of non-linear different transforms on record, MIT-BIH 124
Figure 4.4: Compressed Waveform of record 117 using ENOPV transform at UPRD=1

Figure 4.5: Compressed Waveform of record 117 using ENOPV transform at UPRD=3
Figure 4.6: Compressed Waveform of record 232 using ENOPV transform at

$\text{UPRD}=1$

Figure 4.7: Compressed Waveform of record 232 using ENOPV transform at

$\text{UPRD}=3$
Figure 4.8: Compressed Waveform of record 117 using ENOCA transform at $\text{UPRD}=1$

Figure 4.9: Compressed Waveform of record 117 using ENOCA transform at $\text{UPRD}=3$
Figure 4.10: Compressed Waveform of record 232 using ENOCA transform at

\[ \text{UPRD}=1 \]

Figure 4.11: Compressed Waveform of record 232 using ENOCA transform at

\[ \text{UPRD}=3 \]
Figure 4.12: Compressed Waveform of record 117 using Maxlift transform at $UPRD=1$

Figure 4.13: Compressed Waveform of record 117 using Maxlift transform at $UPRD=3$
Figure 4.14: Compressed Waveform of record 232 using Maxlift transform at UPRD=1

Figure 4.15: Compressed Waveform of record 232 using Maxlift transform at UPRD=3
Figure 4.16: Compressed Waveform of record 117 using Medlift transform at $\text{UPRD}=1$

Figure 4.17: Compressed Waveform of record 117 using Medlift transform at $\text{UPRD}=3$
Figure 4.18: Compressed Waveform of record 232 using Medlift transform at

UPRD=1

Figure 4.19: Compressed Waveform of record 232 using Medlift transform at

UPRD=3
Figure 4.20: Compressed Waveform of record 117 using LWT at UPRD=1

Figure 4.21: Compressed Waveform of record 117 using LWT at UPRD=3
Figure 4.22: Compressed Waveform of record 232 using LWT at UPRD=1

Figure 4.23: Compressed Waveform of record 232 using LWT at UPRD=3
4.4 ECG Compression with Normalization using Non-Linear Transform

The concept of normalization is already explained in chapter 3. Here effect of normalization on nonlinear transform is studied with LWT. In LWT, ECG signal is transformed using cdf 9/7 and decomposed to 5 levels.

The normalization process increases the number of zero coefficients during thresholding which increases the CR is also check with LWT. From the results it can be observed that CR improves with normalization for non linear transformation. It can be verified from Table 4.5 that by using the LWT transformation for the record MIT-BIH 121, CR is 13.17, 28.28, 95.65 and 120.73 for UPRD is 0.5, 1, 2 and 3 respectively without normalization but from Table 4.7 for the same ECG record CR is 13.42, 30.25, 148.12 and 228.46 for the UPRD=0.5, 1, 2, and 3 respectively with normalization. It can also be observed from the results given in tables that effect of normalization in improving the CR is less significant at lower UPRD.

Next, to show the performance of normalization using LWT on different ECG signal (number of samples are 43200, Time=2 min) at different PRDs (ranging from 0.5 to 3), results are presented graphically in Figure 4.24 to Figure 4.26. Here, the non zero coefficients after thresholding are quantized with 12 bit Max-Lloyd quantizer. Figure 4.24 to Figure 4.26 show that CR improve with the introduction of normalization in LWT.

Further, for visual comparison of ECG compression for normalization using LWT are tested at low PRD (1) and high PRD (3) on normal (MIT-BIH 117) rhythm and abnormal (MIT-BIH 232) rhythm. The original and reconstructed signal along with error signal is shown in Figure 4.27- Figure 4.30. The closer look on figures reveals that reconstructed signal is almost identical to original signal.
Table 4.7: Performance of ECG compression with normalization using LWT on different ECG signals

LWT with Normalization, \(^1\)Qbits=12, Samples=43200, Time=2 min, [Aggarwal and Patterh, 2013]

<table>
<thead>
<tr>
<th>Signal</th>
<th>(^{^1})UPRD=0.5</th>
<th>(^{^1})UPRD=1</th>
<th>(^{^1})UPRD=2</th>
<th>(^{^1})UPRD=3</th>
</tr>
</thead>
<tbody>
<tr>
<td>BPRD</td>
<td>QPRD</td>
<td>CR</td>
<td>BPRD</td>
<td>QPRD</td>
</tr>
<tr>
<td>121</td>
<td>0.49</td>
<td>0.49</td>
<td>13.42</td>
<td>1.00</td>
</tr>
<tr>
<td>122</td>
<td>0.50</td>
<td>0.50</td>
<td>12.11</td>
<td>0.99</td>
</tr>
<tr>
<td>233</td>
<td>0.50</td>
<td>0.50</td>
<td>12.81</td>
<td>1.00</td>
</tr>
<tr>
<td>103</td>
<td>0.50</td>
<td>0.50</td>
<td>11.67</td>
<td>1.00</td>
</tr>
<tr>
<td>123</td>
<td>0.50</td>
<td>0.50</td>
<td>12.34</td>
<td>1.00</td>
</tr>
<tr>
<td>116</td>
<td>0.50</td>
<td>0.50</td>
<td>14.56</td>
<td>0.99</td>
</tr>
<tr>
<td>219</td>
<td>0.50</td>
<td>0.50</td>
<td>13.95</td>
<td>0.99</td>
</tr>
<tr>
<td>115</td>
<td>0.50</td>
<td>0.50</td>
<td>11.95</td>
<td>0.99</td>
</tr>
<tr>
<td>223</td>
<td>0.49</td>
<td>0.49</td>
<td>12.53</td>
<td>1.00</td>
</tr>
<tr>
<td>205</td>
<td>0.50</td>
<td>0.50</td>
<td>11.73</td>
<td>0.99</td>
</tr>
</tbody>
</table>

\(^1\)Qbits- bits used for quantization  \(^2\)UPRD- user defined PRD
\(^3\)BPRD- PRD before quantization  \(^4\)QPRD- PRD after quantization  \(^5\)CR-Compression ratio
Figure 4.24: Comparison of ECG compression with normalization and without normalization LWT on record, MIT-BIH 117

Figure 4.25: Comparison of ECG compression with normalization and without normalization LWT on record, MIT-BIH 112
Figure 4.26: Comparison of ECG compression with normalization and without normalization LWT on record, MIT-BIH 124

Figure 4.27: Compressed waveform of record 117 using LWT with normalization at UPRD=1
Figure 4.28: Compressed waveform of record 117 using LWT with normalization at UPRD=3

Figure 4.29: Compressed waveform of record 232 using LWT with normalization at UPRD=1
Figure 4.30: Compressed waveform of record 232 using LWT with normalization at UPRD=3

4.5 Conclusion

In this chapter, impact of various non-linear transforms on ECG compression is studied. From the result one can conclude that, LWT performs better as compared to other non linear and linear transforms at high (3) PRD. At low (0.5) PRD linear transform like DCT still performs better than non-linear transforms. The high performance of quality controlled nonlinear transforms show that these should not be ignored for ECG compression. From Chapter 3 and Chapter 4 it can be concluded that DCT and LWT are best suited for ECG compression. From the results it is also observed that CR improves with normalization for both linear and non linear transformation.