Hindawi Publishing Corporation
EURASIP Journal on Advances in Signal Processing
Volume 2007, Article ID 35641, 20 pages
doi:10.1155/2007/35641

Research Article

Lossless Compression Schemes for ECG Signals Using Neural Network Predictors

R. Kannan and C. Eswaran

Center for Multimedia Computing, Faculty of Information Technology, Multimedia University, Cyberjaya 63100, Malaysia

Received 24 May 2006; Revised 22 November 2006; Accepted 11 March 2007

Recommended by William Allan Sandham

This paper presents lossless compression schemes for ECG signals based on neural network predictors and entropy encoders. Decorrelation is achieved by nonlinear prediction in the first stage and encoding of the residues is done by using lossless entropy encoders in the second stage. Different types of lossless encoders, such as Huffman, arithmetic, and runlength encoders, are used. The performances of the proposed neural network predictor-based compression schemes are evaluated using standard distortion and compression efficiency measures. Selected records from MIT-BIH arrhythmia database are used for performance evaluation. The proposed compression schemes are compared with linear predictor-based compression schemes and it is shown that about 11% improvement in compression efficiency can be achieved for neural network predictor-based schemes with the same quality and similar setup. They are also compared with other known ECG compression methods and the experimental results show that superior performances in terms of the distortion parameters of the reconstructed signals can be achieved with the proposed schemes.

1. INTRODUCTION

Any signal compression algorithm should strive to achieve greater compression ratio and better signal quality without affecting the diagnostic features of the reconstructed signal. Several methods have been proposed for lossy compression of ECG signals to achieve these two essential and conflicting requirements. Some techniques such as the amplitude zone time epoch coding (AZTEC), the coordinate reduction time encoding system (CORTES), the turning point (TP), and the fan algorithm are dedicated and applied only for the compression of ECG signals [1] while other techniques, such as differential pulse code modulation [2–6], subband coding [7, 8], transform coding [9–13], and vector quantization [14, 15], are applied for a wide range of one-, two-, and three-dimensional signals.

Lossless compression schemes are preferable to lossy compression schemes in biomedical applications where even the slight distortion of the signal may result in erroneous diagnosis. The application of lossless compression for ECG signals is motivated by the following factors. (i) A lossy compression scheme is likely to yield a poor reconstruction for a specific portion of the ECG signal, which may be important for a specific diagnostic application. Furthermore, a lossy compression method may not yield diagnostically acceptable results for the records of different arrhythmia conditions. It is also difficult to identify the error range, which can be tolerated for a specific diagnostic application. (ii) In many countries, from the legal point of view, reconstructed biomedical signal after lossy compression cannot be used for diagnosis [16, 17]. Hence, there is a need for effective methods to perform lossless compression of ECG signals. The lossless compression schemes proposed in this paper can be applied to a wide variety of biomedical signals including ECG and they yield good signal quality at reduced compression efficiency compared to the known lossy compression methods.

Entropy encoders are used extensively for lossless text compression but they perform poorly for biomedical signals, which have high correlation between adjacent samples. A two-stage lossless compression technique with a linear predictor in the first stage and a bilevel sequence coder in the second stage is implemented in [2] for seismic data. A method with a linear predictor in the first stage and an

arithmetic coder in the second stage is reported in [18] for seismic and speech waveforms.

Summaries of different ECG compression schemes along with their distortion and compression efficiency performance measures are reported in [1, 14, 15]. A tutorial discussion of predictive coding using neural networks for image compression is given in [3]. Several neural network architectures, such as multilayer perceptron, functional link neural network, and radial basis function network, were investigated for designing a nonlinear vector predictor for image compression and it was shown that they outperform the linear predictors since the nonlinear predictors can exploit higher-order statistics while the linear predictors can exploit only second-order statistics [4].

Performance comparison of several classical and neural network predictors for lossless compression of telemetry data is presented in [5]. Huffman coding and its variations are described in detail in [6] and basic arithmetic coding from the implementation point of view is described in [19]. Improvements on the basic arithmetic coding by using only a small number of multiplicative operations and utilizing low-precision arithmetic are described in [20] which also discusses a modular structure separating the coding, modeling, and probability estimation components of a compression system.

In this paper, we present single- and two-stage compression schemes with multilayer perceptron (MLP) trained with backpropagation learning algorithm as the nonlinear predictor in the first stage followed by Huffman or arithmetic encoders in the second stage for lossless compression of ECG signals. To the best of our knowledge, ECG compression with nonlinear predictors such as neural networks as a decorrelator in the first stage followed by entropy encoders for compressing the prediction residues in the second stage has not been implemented yet. We propose for the first time, compression schemes for ECG signals involving neural network predictors and different types of encoders.

The rest of the paper is organized as follows. In Section 2, we briefly describe the proposed predictor-encoder combination method for the compression of ECG signals along with single- and adaptive-block methods for training the neural network predictor. Experimental setup along with the description of the selected database records are discussed in Section 3 followed by the definition of performance measures used for evaluation in Section 4. Section 5 presents the experimental results and Section 6 shows the performance comparison with other linear predictor-based ECG compression schemes, using selected records from MIT-BIH arrhythmia database [21]. Conclusions are stated in Section 7.

2. PROPOSED LOSSLESS DATA COMPRESSION METHOD

2.1. Description of the method

The proposed lossless compression method is illustrated in Figure 1.

The above lossless compression method is implemented in two different ways, single- and two-stage compression schemes.

In both schemes, a portion of the ECG signal samples is used for training the MLP until the goal is reached. The weights and biases of the trained neural network along with the network setup information are sent to the receiving end for identical network setup. The first p samples are also sent to the receiving end for prediction, where p is the order of prediction. Prediction is done using the trained neural network at the transmitting and receiving ends simultaneously. The residues are generated at the transmitting end, by subtracting the predicted sample values from the target values. In the single-stage scheme, the generated residues are rounded off and sent to the receiving end, where the reconstruction of original samples is done by adding the rounded residues with the predicted samples. In the two-stage schemes, the rounded residues are further encoded with Huffman/arithmetic/runlength encoders in the second stage. The binary-coded residue sequence generated in the second stage is transmitted to the receiving end, where it is decoded in a lossless manner using the corresponding entropy decoder.

The MLP trained with backpropagation learning algorithm is used in the first stage as the nonlinear predictor to predict the current sample using a fixed number, p, of preceding samples. Employing a neural network in the first stage has the following advantages. (i) It exploits the high correlation existing among the neighboring samples of a typical ECG signal, which is a quasiperiodic signal. (ii) It has the inherent properties such as massive parallelism, generalization, error tolerance, flexibility in recall, and graceful degradation which suits the time series prediction applications.

Figure 2 shows the MLP used for the ECG compression which comprises an input layer with p neurons, where p is the order of prediction, a hidden layer with q neurons, and an output layer with a single neuron. In Figure 2, $x_1, x_2, \ldots, x_p$, represent the preceding samples and $\hat{x}_{(p+1)}$ represents the predicted current sample. The residues are generated as shown in (1),

$$r = [x_i - \hat{x}_i], \quad i = p+1, p+2, \ldots, v, \tag{1}$$

where v is the total number of input samples, x_i is the original sample value, and $\hat{x}_i$ is the predicted sample value.

The inputs and outputs for a single hidden layer neuron are as shown in Figure 3. The activation functions used for the hidden layer and the output layer neurons are hyperbolic tangent and linear functions, respectively. The outputs of the hidden and output layers represented as out_{hj} and out_o, respectively, are given by (2) and (3),

$$\text{Out}_{hj} = \text{tansig}\left(\text{Net}_{hj}\right) = \left[\frac{2}{1+\exp\left(-2\text{Net}_{hj}\right)}\right] - 1, \tag{2}$$

where $\text{Net}_{hj} = \sum_{i=1}^{p} w_{ij}x_i + b_j$, $j = 1, \ldots, q$,

$$\text{Out}_o = \text{purelin}\left(\text{Net}_o\right) = \text{Net}_o, \tag{3}$$

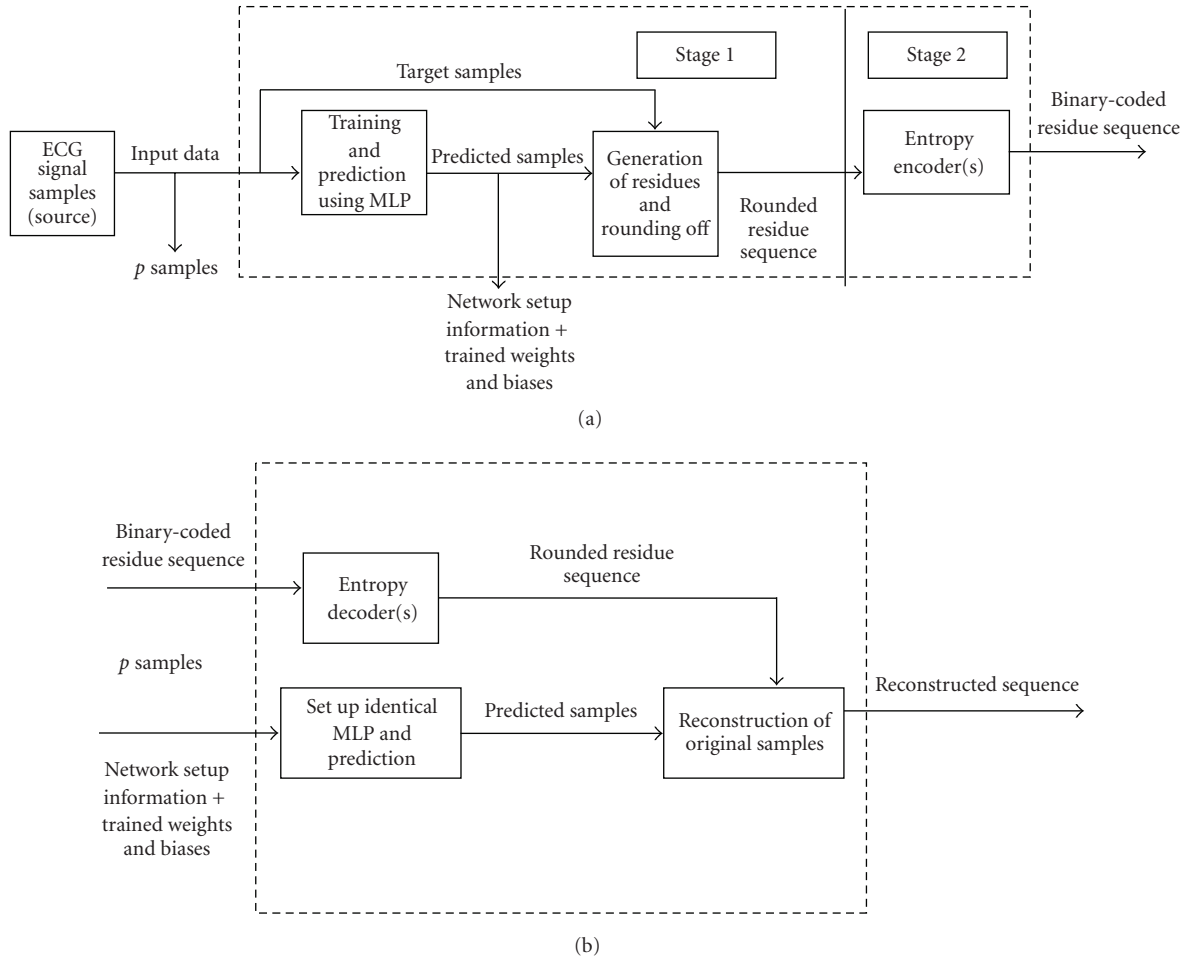

Figure 1: Lossless compression method: (a) transmitting end and (b) receiving end.

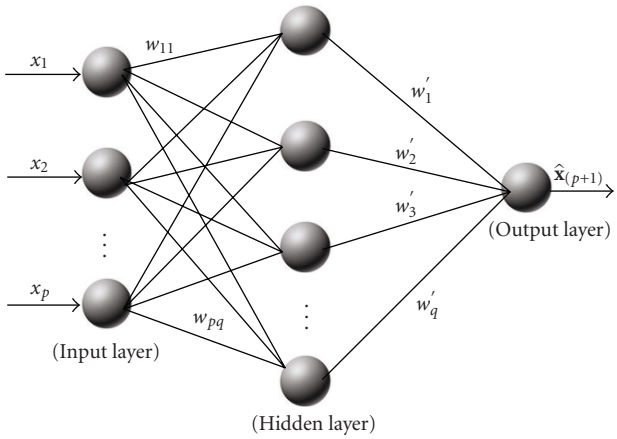

Figure 2: MLP used as a nonlinear predictor.

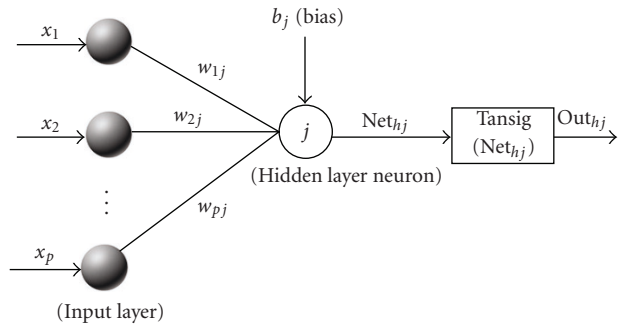

Figure 3: Input and output of a single hidden layer neuron.

where $\text{Net}_o = \sum_{j=1}^{q} \text{out}_{hj} w_j' + b'$, q is the number of hidden layer neurons.

The numbers of input and hidden layer neurons as well as the activation functions are defined based on empirical tests. It was found that the architectural configuration of 4-7-1 with 4 input neurons, 7 hidden layer neurons, and 1 output layer neuron yields the best performance results. With this, we need to send only 35 weights (28 hidden layer and 7 output layer weights) and 8 biases for setting up an identical network configuration at the receiving end. Assuming that 32-bit floating-point representation is used for the weights

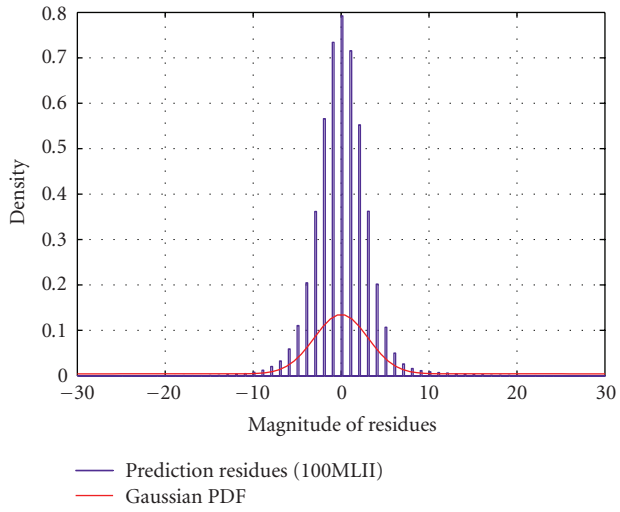

Figure 4: Overlay of Gaussian probability density function over the histogram plot of prediction residues for the MIT-BIH ADB record 100MLII.

and biases, it requires 1376 bits. The MLP is trained with Levenberg-Marquardt backpropagation algorithm [22]. The training goal is to achieve a value of 0.0001 for the mean-squared error between the actual and target outputs. When the specified training goal is reached, the underlying major characteristics of the input signal are stored in the neural network in the form of weights.

The residues generated after prediction are encoded according to the probability distribution of the magnitudes of the residue sequence with Huffman or arithmetic encoders in the second stage. If Huffman or arithmetic coding is used directly without nonlinear predictor in the first stage, the following problems may arise. (i) Huffman or arithmetic coding does not remove the intersample correlation that exists among the neighboring samples of the semiperiodic ECG signal. (ii) The size of the symbol table required for encoding of ECG samples will be too large to be used in any real-time applications.

The histogram of the magnitude of the predicted residue sequence can be approximated by a Gaussian probability density function with most of the prediction residue values concentrated around zero as shown in Figure 4. This figure shows the magnitude of rounded prediction residues for about 216 000 samples after the first stage. As the residue signal has low zero-order entropy compared to the original ECG signal, it can be encoded with lower average bits per sample using lossless entropy coding techniques.

Though the encoder and the decoder used at the transmitting and receiving ends are lossless, the overall two-stage compression schemes can be considered as near-lossless since the residue sequence is rounded off before encoding.

2.2. Training and bit allocation

Two types of methods, namely, single-block training (SBT), and adaptive-block training (ABT) are used for training the MLP [5]. The SBT method, which is used for short-duration ECG signals, makes the transmission faster since the training parameters are transmitted only once to the receiving end to setup the network. The ABT method, which is used for both short- and long-duration ECG signals, can capture the changes in the pattern of the input data, as the input signal is divided into blocks, and the training is performed on each block separately. The ABT method makes the transmission slower because the network setup information has to be sent to the receiving end N times, where N is the number of blocks used.

To begin with, the neural network configuration and the training parameters have to be setup identically on both transmitting and receiving ends. The basic data that have to be sent to the receiving end in the SBT method are the values of the weights, biases, and the first p samples where p is the order of the predictor. If q is the number of neurons in the hidden layer, the number of weights to be sent is $(pq + q)$, where pq and q represent the number of hidden and output layer weights, respectively, and the number of biases to be transmitted is $(q + 1)$, where q and 1 represent the number of hidden and output layer biases, respectively. For ABT method, the above basic data have to be sent for each block after training. The number of samples in each block in the ABT method is determined empirically.

If the training and the network architectural details are not predetermined at the transmitting and receiving ends, the network setup header information have also to be sent in addition to the basic data. We have provided three headers of length 64 bits each in order to send the network architectural information (such as the number of hidden layers, the number of neurons in each hidden layer, and the type of activation functions for hidden and output layers), training information (such as training function, initialization function, performance function, pre- and postprocessing methods, block size, and training window), and training parameters (such as number of epochs, learning rate, performance goal, and adaptation parameters).

The proposed lossless compression schemes are implemented using two different methods. In the first method, the values of the weight, bias, and residues are rounded off and the rounded integer values are represented using 2's complement format. The number of bits required for sending the weight, bias, and residue values are determined as follows:

$$\begin{aligned} w &= \text{ceil}[\log_2(\text{max. absolute weight}) + 1], \\ b &= \text{ceil}[\log_2(\text{max. absolute bias}) + 1], \\ e &= \text{ceil}[\log_2(\text{max. absolute residue}) + 1], \end{aligned} \tag{4}$$

where w is the number of bits used to represent each weight, b is the number of bits used to represent each bias, and e is the number of bits used to represent each residual sample.

In the second method, the residue values are sent in the same format as in the first method but the weights and biases are sent using floating-point representation with 32 or 64 bits. The second method results in identical network setups, at the transmitting and receiving ends.

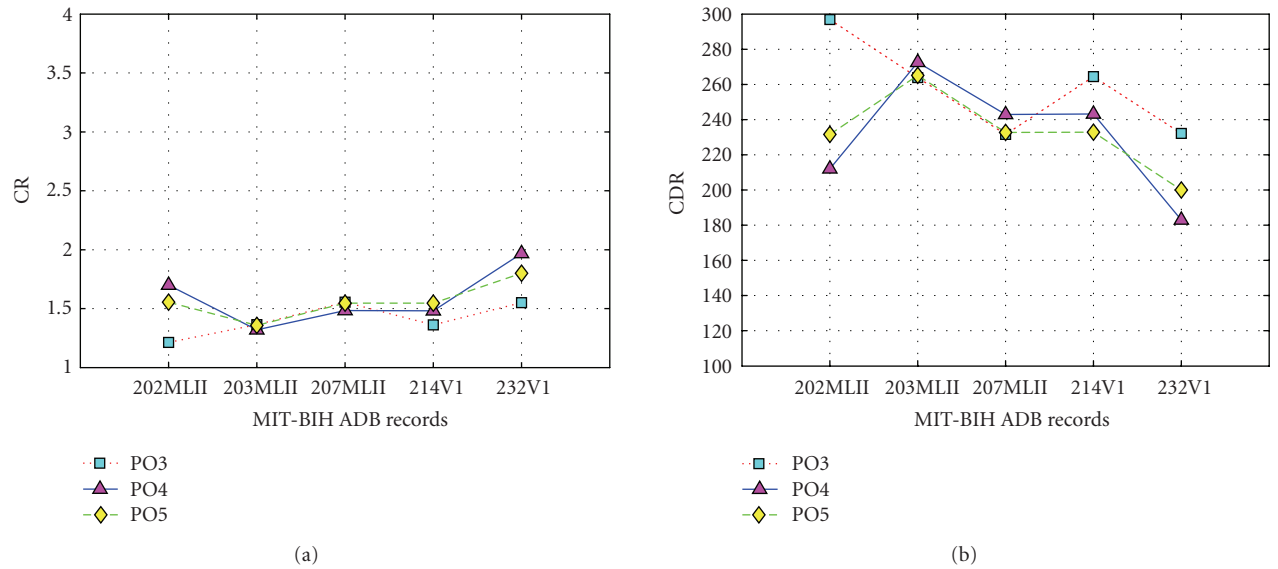

FIGURE 5: Compression efficiency performance results on short-duration datasets with different predictor orders: (a) CR and (b) CDR for P scheme.

For real-time applications, we can use only the prediction stage for compression thereby reducing the overall processing time. This compression scheme will be referred to as the single-stage scheme. For the single-stage compression, the total numbers of bits needed to be sent with the SBT and ABT training methods are given in (5) and (7), respectively,

$$N_{\text{1-stage}}^{\text{SBT}} = N_{bs} + (v - p)e, \tag{5}$$

where $N_{\text{1-stage}}^{\text{SBT}}$ is the number of bits to be sent using SBT method in single-stage compression scheme, v is the total number of input samples, p is the predictor order, and e is the number of bits used to send each residual sample.

N_{bs} is the number of basic data bits that have to be sent for identical network setup at the receiving end,

$$N_{bs} = (pn) + (N_w w) + (N_b b) + (N_{so}), \tag{6}$$

where n is the number of bits used to represent input samples (resolution), N_w is the total number of hidden and output layer weights, N_b is the total number of hidden and output layer biases, w is the number of bits used to represent each weight, b is the number of bits used to represent each bias, and N_{so} is the number of bits used for the network setup overhead,

$$N_{\text{1-stage}}^{\text{ABT}} = (N_{ab}N_{bs}) + [v - (N_{ab}p)]e, \tag{7}$$

where $N_{\text{1-stage}}^{\text{ABT}}$ is the number of bits to be sent using ABT method in a single-stage compression scheme and N_{ab} is the number of adaptive blocks.

The total numbers of bits required for the two-stage compression schemes with the SBT and ABT training methods are given in (8) and (9), respectively,

$$N_{\text{2-stage}}^{\text{SBT}} = N_{bs} + (v - p)R + L_{\text{len}}, \tag{8}$$

where $N_{\text{2-stage}}^{\text{SBT}}$ is the number of bits to be sent using the SBT method in two-stage compression schemes, R is the average code word length obtained for Huffman or arithmetic encoding, and L_{len} represents the bits needed to store Huffman table information. For arithmetic coding, L_{len} is zero,

$$N_{\text{2-stage}}^{\text{ABT}} = (N_{ab}N_{bs}) + [v - (N_{ab}p)](R + L_{\text{len}}), \tag{9}$$

where $N_{\text{2-stage}}^{\text{ABT}}$ is the number of bits to be sent using ABT method in two-stage compression schemes.

2.3. Computational time and cost

In the single-stage compression scheme, once the training is completed at the transmitting end, the basic setup information is sent to the receiving end so that the prediction is done in parallel at both ends. Prediction and generation of residues can be done in sequence for each sample at the transmitting end and the original signal can be reconstructed at the receiving end as the residues are received. Total processing time includes the following time delays: (i) time required for transmitting the basic setup information such as the weights, biases, and the first p samples, (ii) time required for performing the prediction at the transmitting and receiving ends in parallel, (iii) time required for the generation and transmission of residues, and (iv) time required for the reconstruction of original samples.

The computational time required for performing the prediction of each sample depends on the number of multiplication and addition operations required. In this setup, it requires only 28 and 7 multiplication operations at the hidden and output layers, respectively, in addition to the operations required for applying the tangent sigmoid functions for the seven hidden layer neurons and for applying a linear function for the output layer neuron. One subtraction and one

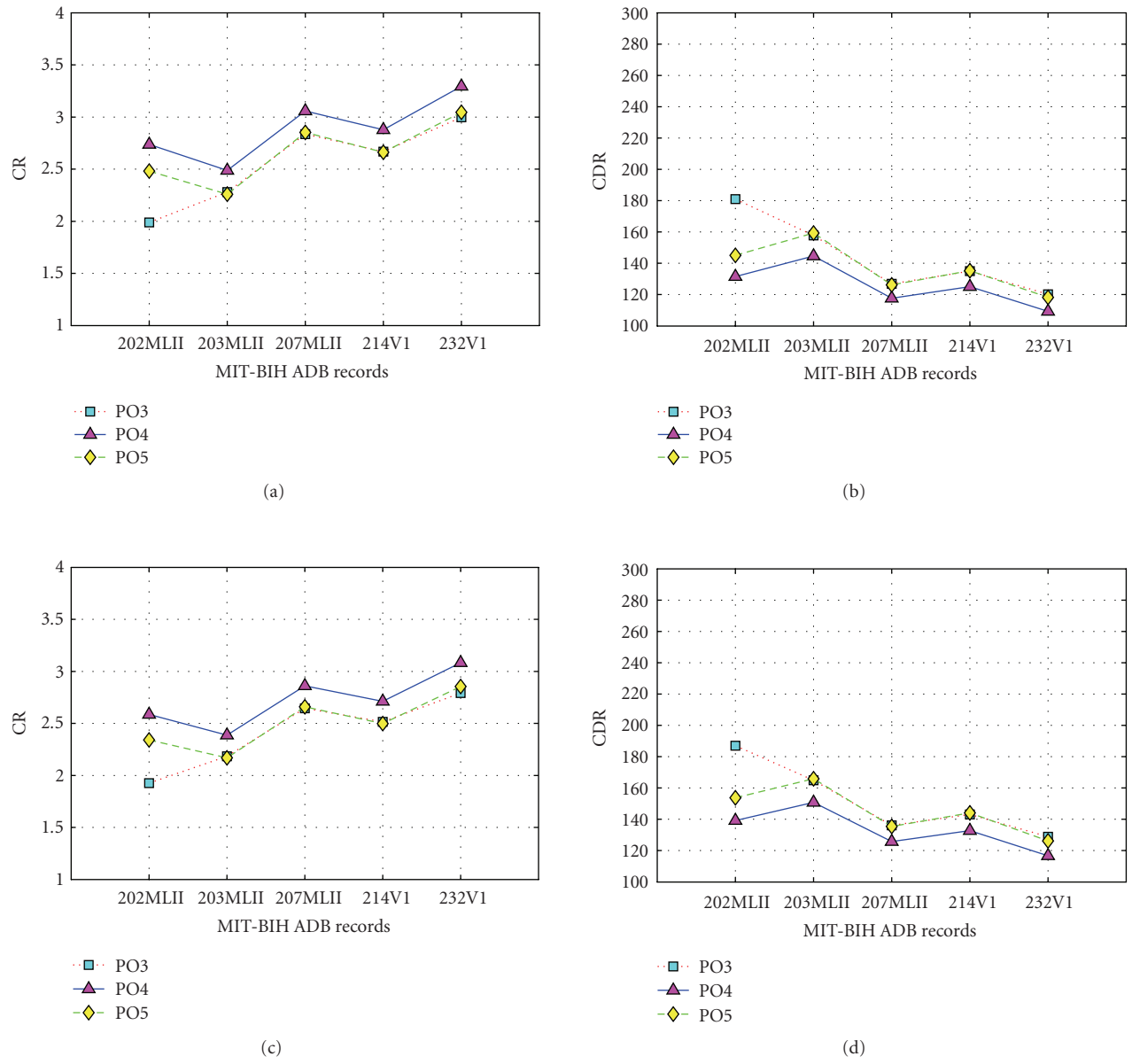

Figure 6: Compression efficiency performance results on short-duration datasets with different predictor orders: (a) CR and (b) CDR for PH scheme, (c) CR and (d) CDR for PRH scheme.

addition operations are required for generating each residue and each reconstructed sample, respectively. As the processing time involved is not significant, this scheme can be used for real-time transmission applications once the training is completed.

The training time depends on the training algorithm used, the number of samples in the training set, the numbers of weights and biases, the maximum number of epochs or the error goal set, and the initial weights. In the proposed schemes, Levenberg-Marquardt algorithm [22] is used since it is considered to be the fastest among the backpropagation algorithms for function approximation if less numbers of weights and biases are used [23]. For the ABT method, 4320 and 1440 samples are used for each block during the training with the first and second datasets, respectively. For the SBT method, 4320 samples are used during the training with the second dataset. The maximum number of epochs and the goal set for both methods are 5000 and 0.0001, respectively.

For the two-stage compression schemes, the time required for encoding and decoding the residues at the transmitting and receiving ends, respectively, should also be taken into account.

3. EXPERIMENTAL SETUP

The proposed compression schemes are tested on selected records from the MIT-BIH arrhythmia database [21]. The

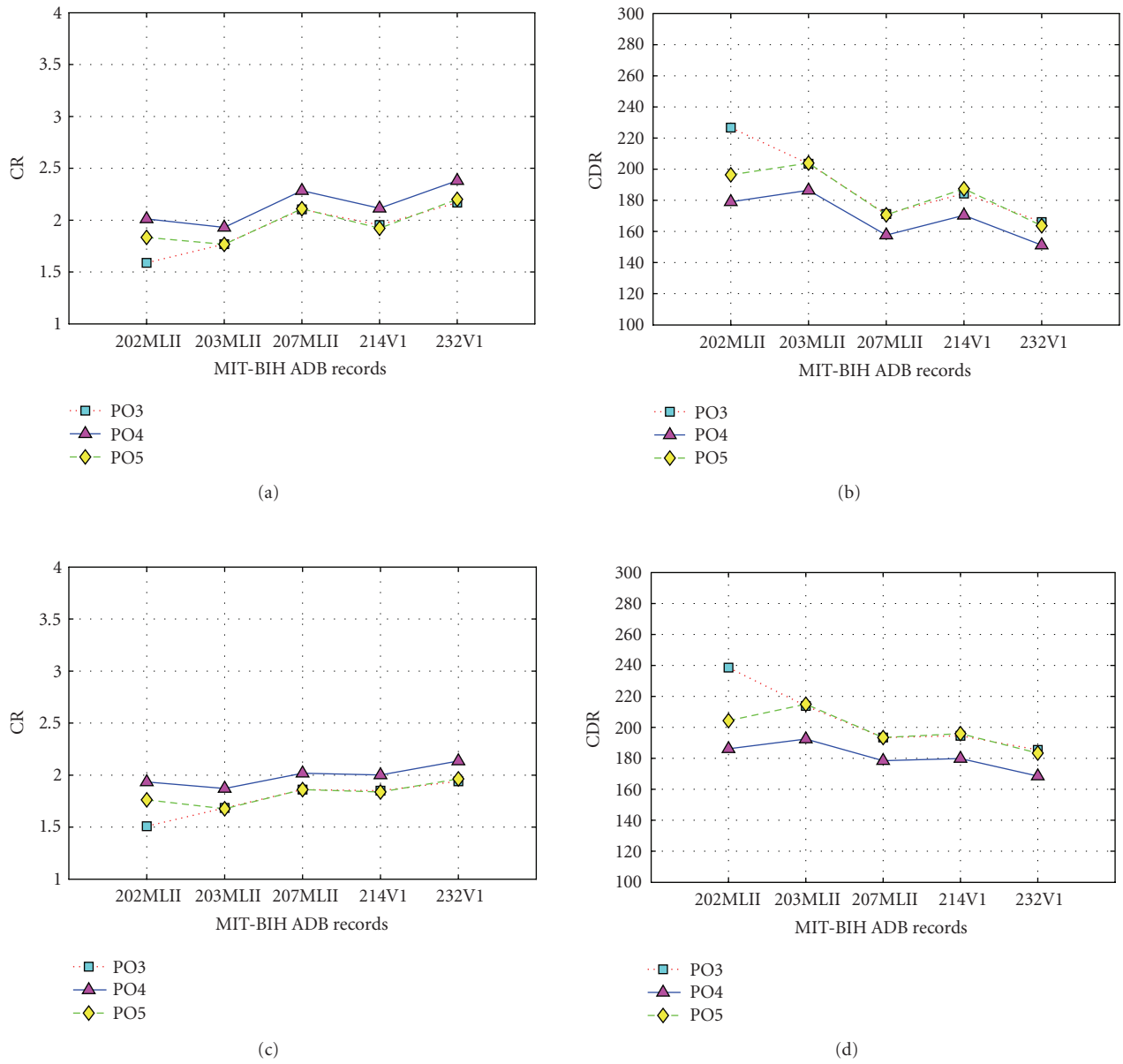

Figure 7: Compression efficiency performance results on short-duration datasets with different predictor orders: (a) CR and (b) CDR for PA scheme, (c) CR and (d) CDR for PRA scheme.

records are selected based on different clinical rhythms aiming at performing the comparison of the proposed schemes with other known compression methods. The selected records are divided into two sets: 10 minutes of ECG samples from the records 100MLII, 117MLII, and 119MLII form the first dataset while 1 minute of ECG samples from the records 202MLII, 203MLII, 207MLII, 214V1, and 232V1 form the second dataset. The data are sampled at 360 Hz where each sample is represented by 11 bits, packed into 12 bits for storage, over a 10 mV range [21].

The MIT-BIH arrhythmia database contains two-channel ambulatory ECG recordings, obtained usually from modified leads, MLII and V1. Normal QRS complexes and ectopic beats are prominent in MLII and V1, respectively. Since the physical activity causes significant interference in the standard limb leads for long-term ECG recordings, modified leads were used and placed in positions so that the signals closely match the standard limb leads. Signals from the first dataset represent the variety of waveforms and artifacts encountered in routine clinical use since they are chosen from the random set. Signals from the second dataset represent complex ventricular, junctional, and supraventricular arrhythmias and conduction abnormalities [21].

The compression performances of the proposed schemes are evaluated with the long-duration signals (i.e., the first dataset comprising 216 000 samples) only for the ABT method. With the short-duration signals (i.e., second dataset comprising 21 600 samples), the performances are evaluated

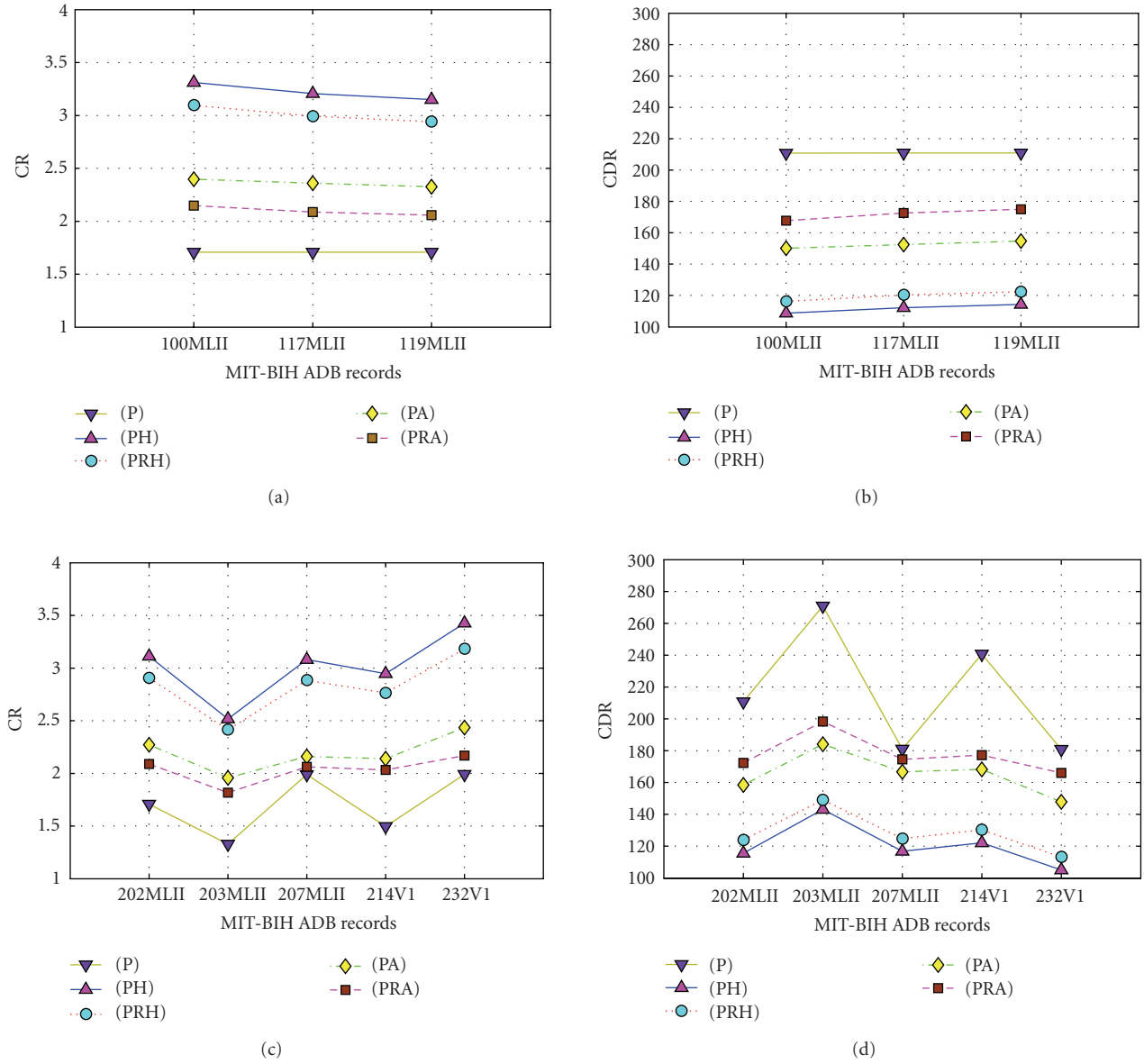

Figure 8: Compression efficiency performance results for different compression schemes: (a) CR and (b) CDR using ABT on long-duration dataset, (c) CR and (d) CDR using SBT on short-duration dataset.

for both SBT and ABT methods. For the ABT method, the samples of the first dataset are divided into ten blocks with 21 600 samples in each block, while the samples of the second dataset are divided into three blocks with 7200 samples in each block. For the SBT method, the entire samples of the second dataset are treated as a single block. The number of blocks used in ABT, and the percentage of samples used for training and testing in the ABT and SBT are chosen empirically.

4. PERFORMANCE MEASURES

An ECG compression algorithm should achieve good reconstructed signal quality for preserving the diagnostic features of the signal and high compression efficiency for reducing the storage and transmission requirements. The distortion measures, such as percent of root-mean-square difference (PRD), root-mean-square error (RMS), and signal-to-noise ratio (SNR), are widely used in the ECG data compression literature to quantify the quality of the reconstructed signal compared to the original signal. The performance measures, such as bits per sample (BPS), compressed data rate (CDR) in bit/s, and compression ratio (CR), are widely used to determine the redundancy reduction capability of an ECG compression method. The proposed compression methods are evaluated using the above standard measures to perform comparison with other methods. Interpretation of results from different compression methods requires careful

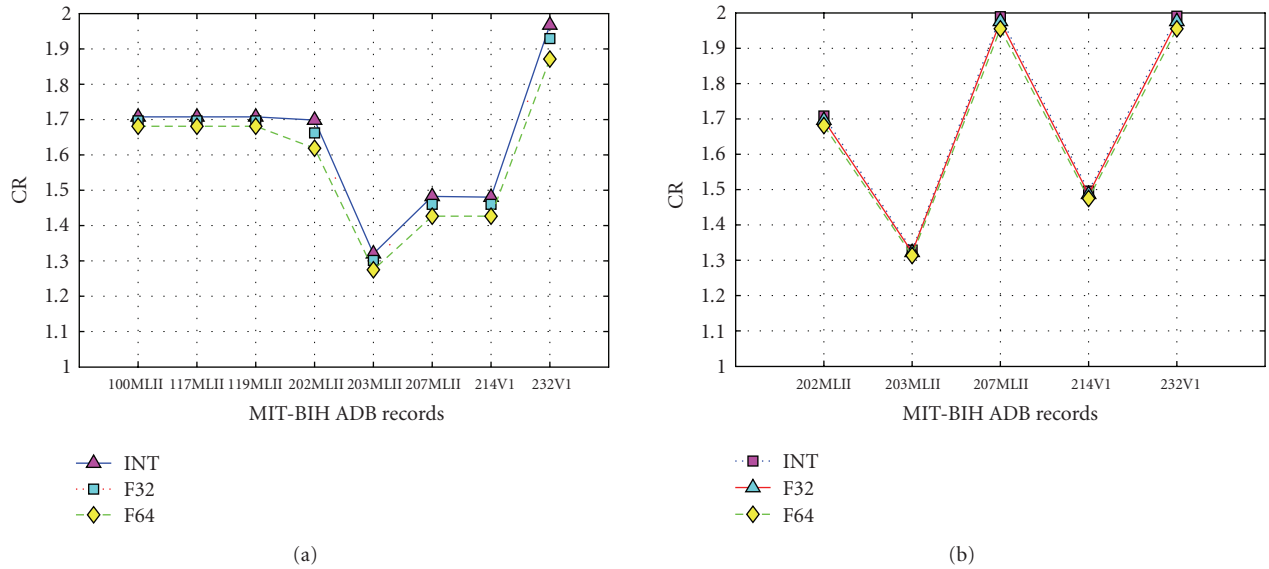

FIGURE 9: Results with floating-point and fixed-point representations for the trained weights and biases for P scheme using (a) ABT on long- and short-duration datasets and (b) SBT on the short-duration dataset. INT, signed 2's complement for representing the weights and biases. F32, 32-bit floating point for representing the weights and biases. F64, 64-bit floating point for representing the weights and biases.

evaluation and comparison, since the database used by different methods may be digitized with different sampling frequencies and quantization bits.

4.1. Distortion measures

4.1.1. Percent of root-mean-square difference and normalized PRD

The PRD is the most commonly used distortion measure in the literature since it has the advantage of low computational complexity.

PRD is defined as [24]

$$\mathrm{PRD} = 100\sqrt{\frac{\sum_{n=1}^{N}\left(x(n)-\hat{x}(n)\right)^2}{\sum_{n=1}^{N}x^2(n)}}, \tag{10}$$

where $x(n)$ is the original signal, $\hat{x}(n)$ is the reconstructed signal, and N is the length of the window over which the PRD is calculated.

If the selected signal has baseline fluctuations, then the variance of the signal will be higher and the PRD will be artificially lower [24]. Therefore, to eliminate the error due to DC level of the signal, a normalized PRD denoted as NPRD can be used [24],

$$\mathrm{NPRD} = 100\sqrt{\frac{\sum_{n=1}^{N}\left(x(n)-\hat{x}(n)\right)^2}{\sum_{n=1}^{N}\left(x(n)-\overline{x}\right)^2}}, \tag{11}$$

where $\overline{x}$ is the mean of the signal.

4.1.2. Root-mean-square error

The RMS is defined as [25]

$$\mathrm{RMS} = \sqrt{\frac{\sum_{n=1}^{N}\left(x(n)-\hat{x}(n)\right)^2}{N}}, \tag{12}$$

where N is the length of the window over which reconstruction is done.

4.1.3. Signal-to-noise ratio and normalized SNR

The SNR is defined as

$$\mathrm{SNR} = 10\log_{10}\left[\frac{\sum_{n=1}^{N}x^2(n)}{\sum_{n=1}^{N}\left(x(n)-\hat{x}(n)\right)^2}\right]. \tag{13}$$

The NSNR as defined in [24, 25] is given by

$$\mathrm{NSNR} = 10\log_{10}\left[\frac{\sum_{n=1}^{N}\left(x(n)-\overline{x}\right)^2}{\sum_{n=1}^{N}\left(x(n)-\hat{x}(n)\right)^2}\right]. \tag{14}$$

The relation between NSNR and NPRD [26] is given by

$$\mathrm{NSNR} = 40 - \left[20\log_{10}(\mathrm{NPRD})\right]\ \mathrm{dB}. \tag{15}$$

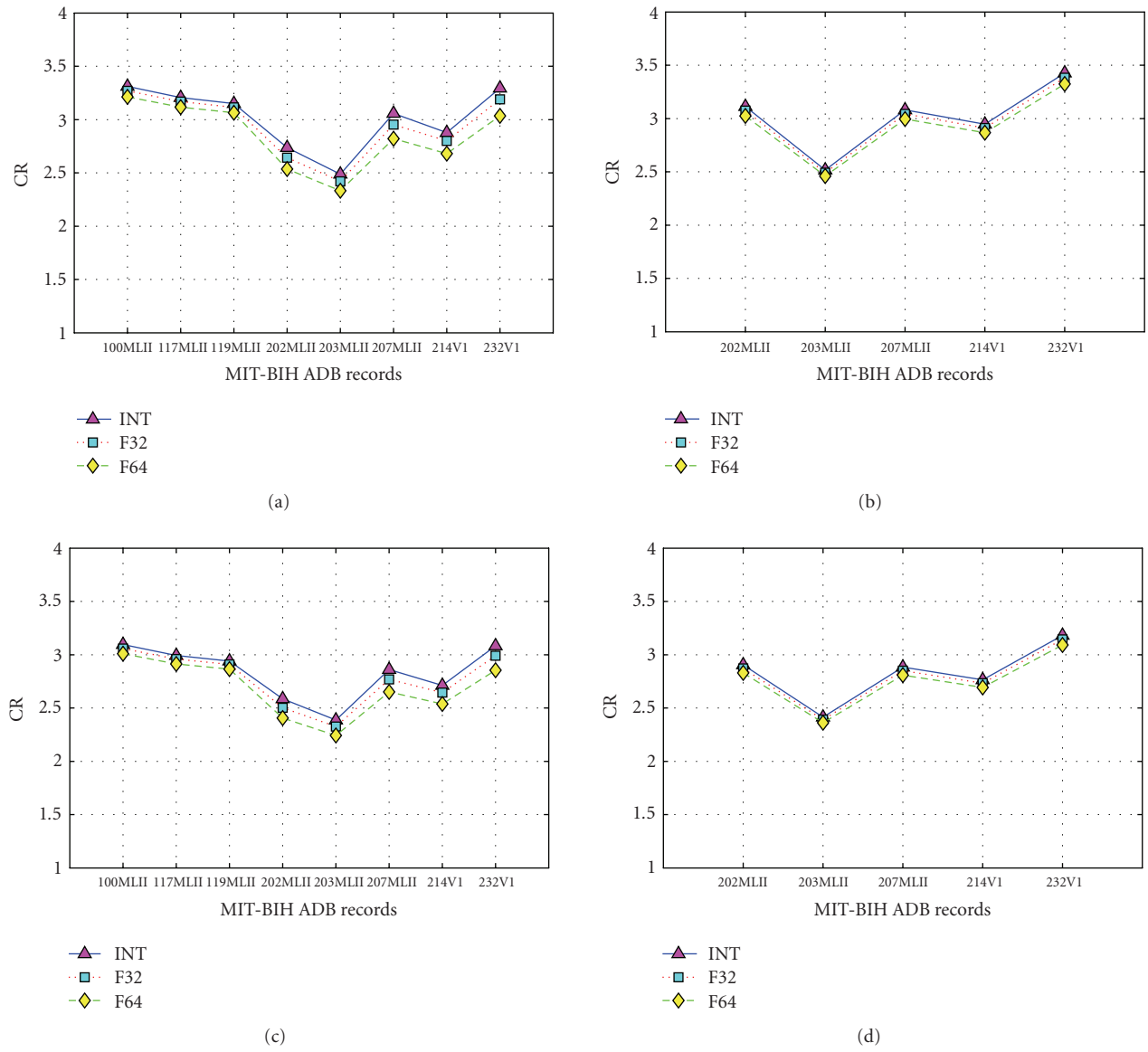

Figure 10: Results with floating-point and fixed-point representations for the trained weights and biases with PH scheme using (a) ABT and (b) SBT; and with PRH scheme using (c) ABT and (d) SBT.

The relation between SNR and PRD [26] is given by

$$\text{SNR} = 40 - [20\log_{10}(\text{PRD})]\ \text{dB}. \tag{16}$$

4.2. Compression efficiency measures

4.2.1. Bits per sample

BPS indicates the average number of bits used to represent one signal sample after compression [6],

$$\text{BPS} = \frac{\text{number of bits required after compression}}{\text{total number of input samples}}. \tag{17}$$

4.2.2. Compressed data rate in bit/s

CDR can be defined as [15]

$$\text{CDR} = \frac{(f_s B_{\text{total}})}{L}, \tag{18}$$

where f_s is the sampling rate, B_{total} is the total number of compressed bits to be transmitted or stored, and L is the data size.

4.2.3. Compression ratio

CR can be defined as [10]

$$\text{CR} = \frac{\text{total number of bits used in the original signal}}{\text{total number of bits used in the compressed signal}}. \tag{19}$$

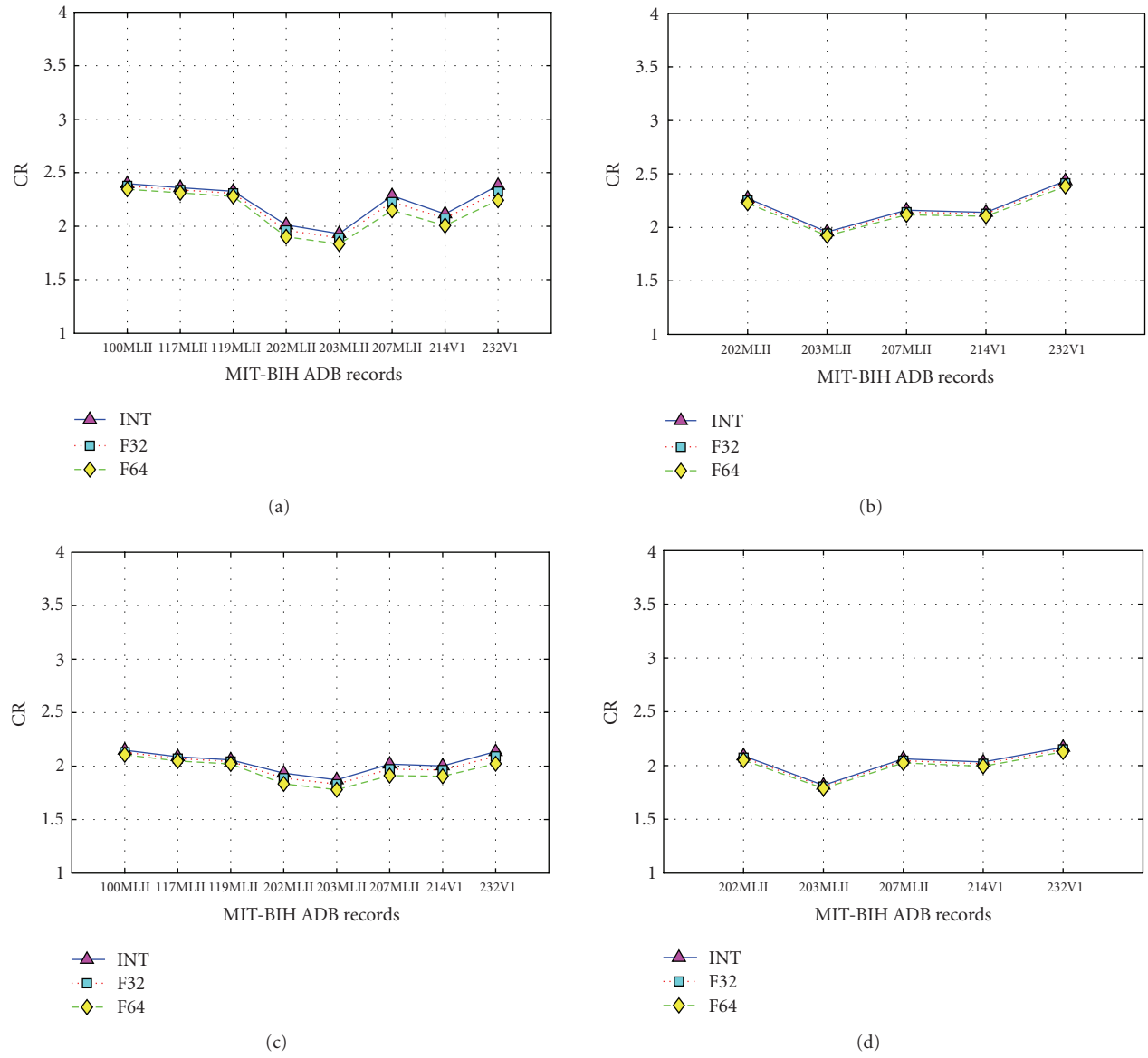

Figure 11: Results with floating-point and fixed-point representations for the trained weights and biases with PA scheme using (a) ABT and (b) SBT; and with PRA scheme using (c) ABT and (d) SBT.

5. RESULTS AND DISCUSSION

We have evaluated the quality and compression efficiency performances of the following five schemes using ABT and SBT training methods: (i) single-stage scheme with MLP as the predictor (denoted as P); (ii) two-stage scheme with MLP predictor in the first stage and Huffman encoder in the second stage (denoted as PH); (iii) two-stage scheme with MLP predictor in the first stage followed by runlength and Huffman encoders in the second stage (denoted as PRH); (iv) two-stage scheme with MLP predictor in the first stage and arithmetic encoder in the second stage (denoted as PA); (v) two-stage scheme with MLP predictor in the first stage followed by runlength and arithmetic encoders in the second stage (denoted as PRA).

5.1. *Distortion and compression efficiency performance results*

The values of the distortion measures obtained using the ABT method on short-duration dataset with a third-order (PO3), fourth-order (PO4), and fifth-order (PO5) predictor are given in Table 1. It should be noted that the distortion measures remain the same for a particular record, irrespective of the type of encoder used in the second stage, since the residues are losslessly encoded for all the two-stage schemes.

From Table 1, it can be noted that the quality measures for all the tested records do not differ significantly with different predictor orders. Hence, the selection of a predictor order can be based on the compression efficiency measures.

Table 1: Quality performance results using ABT method on short-duration dataset with different predictor orders.

Distortion measure	Predictor order	ABT method on the second dataset				
		202MLII	203MLII	207MLII	214V1	232V1
PRD (%)	PO3	0.0245	0.0289	0.0288	0.0276	0.0280
	PO4	0.0290	0.0288	0.0287	0.0276	0.0279
	PO5	0.0289	0.0291	0.0287	0.0276	0.0278
NPRD (%)	PO3	0.4968	0.2789	0.4007	0.2778	0.8830
	PO4	0.5884	0.2776	0.3993	0.2781	0.8798
	PO5	0.5856	0.2800	0.3985	0.2775	0.8779
SNR (dB)	PO3	72.2123	70.7702	70.8098	71.1824	71.0614
	PO4	70.7520	70.8122	70.8424	71.1818	71.0879
	PO5	70.7819	70.7343	70.8568	71.1915	71.1128
NSNR (dB)	PO3	46.0771	51.0914	47.9436	51.1245	41.0812
	PO4	44.6065	51.1316	47.9740	51.1160	41.1123
	PO5	44.6481	51.0572	47.9907	51.1355	41.1313
RMS	PO3	0.2444	0.2888	0.2890	0.2890	0.2891
	PO4	0.2894	0.2874	0.2879	0.2893	0.2880
	PO5	0.2880	0.2899	0.2873	0.2886	0.2873

Table 2: Quality performance results using ABT and SBT methods with a fourth-order predictor.

Distortion measure	ABT method on the first dataset			SBT method on the second dataset				
	100MLII	117MLII	119MLII	202MLII	203MLII	207MLII	214V1	232V1
PRD (%)	0.0301	0.0337	0.0336	0.0289	0.0289	0.0287	0.0276	0.0281
NPRD (%)	0.8078	0.6044	0.2706	0.5863	0.2787	0.3994	0.2775	0.8863
SNR (dB)	70.4287	69.4474	69.4732	70.7820	70.7820	70.8424	71.1818	71.0259
NSNR (dB)	41.8539	44.3735	51.3534	44.6376	51.0973	47.9718	51.1347	41.0484
RMS	0.2893	0.2888	0.2888	0.2884	0.2886	0.2880	0.2887	0.2901

The values of the corresponding compression efficiency results obtained using ABT method for all the compression schemes with different predictor orders are shown in Figures 5–7.

From Figures 5–7, it can be concluded that fourth-order neural network predictor produces better compression efficiency results with the selected records for all the two-stage compression schemes. Hence, with a fourth-order predictor, we have tested the ABT and SBT methods on the long- and short-duration datasets, respectively. The quality performance results obtained with both methods are shown in Table 2.

The values of the corresponding compression efficiency measures are shown in Figure 8.

The results shown are calculated by assuming that the weights, biases, and residues are sent as rounded signed integers in 2's complement format.

From Tables 1 and 2, it is observed that the quality performances do not differ significantly for the different records. Hence, it can be concluded that the proposed methods can be used for a wide variety of ECG data with different clinical rhythms and QRS morphological characteristics. By applying the ABT and SBT methods on the second dataset, it is also observed from Tables 1 and 2 that the quality performance is almost the same for both methods. However, it is clear from the results shown in Figures 5–8 that SBT has superior compression performance compared to ABT for short-duration signals.

From Figures 5–8, it is also observed that the two-stage compression schemes give better compression performance results compared to single-stage compression scheme, while the quality performance results are the same for both methods. Among the different two-stage compression schemes, the PH scheme, using MLP predictor in the first stage and the Huffman encoder in the second stage, gives the best result.

The average number of bits per sample required for arithmetic coding is higher compared to Huffman coding since the prediction of ECG signal yields large number of residues with different magnitudes and unbiased probability distributions. It is also observed that using the runlength encoding on the prediction residues before applying either Huffman or arithmetic coding not only increases the complexity, but also results in slight degradation in compression efficiency performance. This is because of the nature of the residues which may not be suitable for applying the

Table 3: ABT method: comparison between the theoretical bounds and the results obtained using Huffman encoder.

Measure	First dataset			Second dataset				
	100MLII	117MLII	119MLII	202MLII	203MLII	207MLII	214V1	232V1
LB	3.5612	3.6906	3.7566	4.2797	4.6881	3.8149	4.0183	3.5036
UB (Gallagher)	3.8057	3.9156	3.9791	4.4784	4.8585	4.0398	4.2129	3.7481
R	3.5955	3.7112	3.7777	4.3119	4.7258	3.8349	4.0557	3.5356
BPS	3.6249	3.7425	3.8087	4.3860	4.8224	3.9237	4.1718	3.6414

Table 4: SBT method: comparison between the theoretical bounds and the results obtained using Huffman encoder.

Measure	Second dataset				
	202MLII	203MLII	207MLII	214V1	232V1
LB	3.8008	4.7140	3.8390	4.0042	3.4387
UB (Gallagher)	4.0251	4.8751	4.0465	4.1976	3.6857
R	3.8251	4.7344	3.8660	4.0434	3.4697
BPS	3.8550	4.7698	3.8957	4.0733	3.5023

runlength encoding. Furthermore, runlength encoding increases the number of residues to be transmitted to the receiving end.

The compression efficiency results obtained for the proposed compression schemes using 32- and 64-bit floating-point representation for weights and biases are compared with the results obtained using signed 2's complement representation and they are shown in Figures 9–11.

Figures 9–11 confirm that by using 32-bit or 64-bit floating-point representation for the trained weights and biases to setup identical network at the receiving end, the reduction in compression efficiency performance is not significant.

5.2. Theoretical bounds and actual results

According to Shannon's theorem of coding [27], it is impossible to encode the messages generated randomly from a model using less number of bits than the entropy of that model. Lower and upper bounds of compression rates for Huffman encoding denoted as R_{huff} can be given as follows [27]:

$$h(p) \leq R_{\text{huff}} \leq h(p) + 1, \tag{20}$$

where $h(p)$ is the zero-order entropy of the signal.

Gallagher [28] provides the alternative upper bound as follows:

$$R_{\text{huff}} \leq h(p) + p_{\max} + 0.086, \tag{21}$$

where $P_{\max}$ is the maximum probability of a sample.

Tables 3 and 4 give the comparison between the theoretical and actual results for the ABT and SBT methods, respectively. In these tables, R refers to the average code word length obtained with Huffman encoder, and LB and UB refer to the lower and upper theoretical bounds, respectively. The R value is calculated based on the average number of bits required for sending the residues to the receiving end. The BPS value is calculated based on R, and the basic data such as weights, biases, and first p samples to be sent to the receiving end for identical network setup.

From Tables 3 and 4, we can conclude that the average code word lengths for the Huffman encoder are much closer to the theoretical lower bounds, thus obtaining an optimal compression rate. It can also be noted that the values for the BPS falls between lower and upper bounds provided by Gallagher [28] for all the records.

5.3. Relationship between the mean values and compression performance

The performance of the proposed compression schemes may depend upon the characteristics of the input samples also. Table 5 shows the PRD and CR values and the corresponding mean values of the records. The mean values are calculated after subtracting the offset of 1024 from each sample. From Table 5, it is noted that for most of the records, higher mean values result in lower PRD and higher CR values. This result is obtained by treating the first and second datasets independently.

5.4. Visual inspection

Figures 12–15 show the original, reconstructed, and the residue signals, respectively, for about 6 seconds from four records of the MIT/BIH arrhythmia database.

From the visual inspection of the above figures, it can be noted that there is a small variation in the magnitude of the residue signal, irrespective of the fluctuations in the original signal, for different records. Hence, it can be concluded that the proposed compression schemes can be applied for ECG records with different rhythms and QRS morphological characteristics.

Table 5: Relationship between the mean values of the signal and performance of the proposed compression schemes.

Performance measure	First dataset			Second dataset				
	119MLII	117MLII	100MLII	203MLII	202MLII	207MLII	214V1	232V1
Mean	−171.058	−168.792	−63.286	−31.528	−28.417	−23.589	8.495	18.035
PRD (%)	0.034	0.034	0.030	0.029	0.029	0.029	0.028	0.028
CR	3.151	3.206	3.310	2.516	3.100	3.100	2.945	3.426

Table 6: Quality performance results: comparison between NNP- and LP-based compression schemes on short-duration datasets. NNP: neural network predictor, WLSE denotes weighted least-squares error linear predictor, and MSE denotes mean-square error linear predictor.

MIT-BIH ADB record	Type of predictor	PRD (%)	SNR (dB)	NPRD (%)	NSNR (dB)	RMS
202MLII	NNP	0.0289	70.7820	0.5863	44.6376	0.2884
	WLSE	0.0288	70.8231	0.5829	44.6882	0.2867
	MSE	0.0287	70.8294	0.5825	44.6944	0.2865
203MLII	NNP	0.0289	70.7820	0.2787	51.0973	0.2886
	WLSE	0.0290	70.7574	0.2793	51.0790	0.2892
	MSE	0.0290	70.7595	0.2792	51.0810	0.2891
207MLII	NNP	0.0287	70.8424	0.3994	47.9718	0.2880
	WLSE	0.0287	70.8496	0.3990	47.9808	0.2876
	MSE	0.0289	70.7864	0.4019	47.9176	0.2897
214V1	NNP	0.0276	71.1818	0.2775	51.1347	0.2887
	WLSE	0.0276	71.1715	0.2782	51.1137	0.2894
	MSE	0.0275	71.2235	0.2765	51.1657	0.2876
232V1	NNP	0.0281	71.0259	0.8863	41.0484	0.2901
	WLSE	0.0276	71.1943	0.8696	41.2132	0.2847
	MSE	0.0281	71.0393	0.8853	41.0582	0.2898

Table 7: Improvement percentage of NNP-based over LP-based compression schemes using average CR values.

Compression scheme	NNP over WLSE (%)	NNP over MSE (%)
PH	9.5324	14.3198
PRH	9.4708	13.4358
PA	8.6865	12.7675
PRA	8.8861	11.1478

6. PERFORMANCE COMPARISON WITH OTHER METHODS

6.1. Comparison with linear predictor-based compression methods

We have implemented the compression of ECG signals based on two standard linear predictor (LP) algorithms, weighted least-squares error (WLSE), and mean-square error (MSE), for performance comparison with the proposed nonlinear neural network predictor (NNP)-based ECG compression schemes. LP algorithms are based on designing a filter that produces an estimate of the current sample using a linear combination of the past samples such that the cost function such as the WLSE or MSE is minimized. The implementation of WLSE algorithm is based on the adaptive adjustment of the filter coefficients at each instant of time by minimizing the weighted sum of the prediction error. The identified filter coefficients are used to predict the current sample [29, 30]. The implementation of MSE algorithm is based on the Levinson-Durbin recursive method for computing the coefficients of the prediction-error filter of order p, by solving the Wiener-Hopf equations and minimizing the mean-square prediction error [30]. In both WLSE and MSE algorithms, fourth-order predictor is used to compare with the fourth-order NNP-based compression schemes. Two-stage ECG compression schemes are implemented with a WLSE or MSE predictor in the first stage for performance comparison with the proposed schemes.

Table 6 shows the quality performance results of NNP- and LP-based compression schemes using SBT method on short-duration datasets. It should be noted that the quality performance results remain the same for a particular record irrespective of the type of lossless encoder used in the second stage.

From Table 6, it can be concluded that there is a small difference in the quality performance results of NNP- and LP-based compression schemes for a particular record. Figures 16 and 17 show the comparison of compression efficiency performance results between NNP- and LP-based two-stage compression schemes using SBT method on short-duration datasets.

Table 8: Record MIT-ADB 100: performance comparison results with different ECG compression methods. PH denotes MLP predictor in the first stage and Huffman encoder in the second stage. WTDVQ denotes wavelet transform coding using dynamic vector quantization. OZWC denotes optimal zonal wavelet coding. WHOSC denotes wavelet transform higher-order statistics-based coding. CAB denotes cut and align beats approach. TSVD denotes truncated singular-value decomposition. WPC denotes wavelet packet-based compression.

Measure	Proposed scheme (PH)	(WTDVQ) [15]	(OZWC) [9]	(WHOSC) [9]	(CAB) [10]	(TSVD) [31]	(WPC) [32]
PRD (%)	0.0301	6.6	0.5778	1.7399	1.9	9.92	11.58
CR	3.3104	—	8.16	17.51	4.0	77.91	23.61
CDR	108.7474	91.3	—	—	—	50.8	167.7
BPS	3.6249	—	1.47	0.68	—	—	—

Table 9: Record MIT-ADB 117: performance comparison results with different ECG compression methods. DWTC denotes discrete wavelet transform-based coding. FPWCZ denotes fixed percentage of wavelet coefficients to be zeroed. SPIHT denotes set partitioning in hierarchical trees algorithm. MEZWC denotes modified embedded zero-tree wavelet coding. DSWTC denotes discrete symmetric wavelet transform coding.

Measure	Proposed scheme (PH)	(DWTC) [11, 12]	(FPWCZ) [33]	(WTDVQ) [15]	(SPIHT) [34]	(TSVD) [31]	(MEZWC) [35]	(DSWTC) [13]
PRD (%)	0.0337	0.473	2.5518	3.6	1.18	1.18	2.6	3.9
NPRD(%)	0.6044	8.496	—	—	—	—	—	—
CR	3.2064	10.9	16.24	—	8.0	10.0	8.0	8.0
CDR	112.276	—	—-	101.6	—	—	—	—

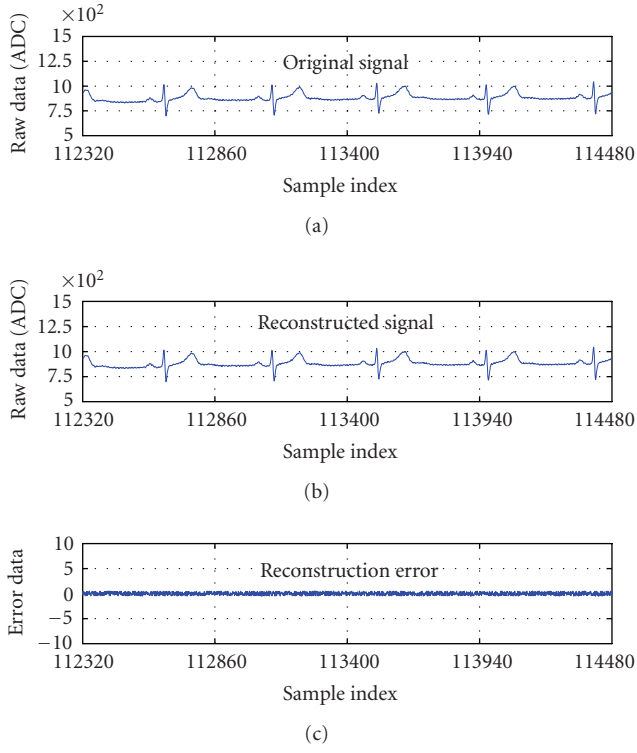

Figure 12: ABT method: original ECG signal along with the reconstructed and residue signals of MIT-BIH ADB record *117MLII*. In the reconstructed signal, PRD = 0.0337, NPRD = 0.6044, SNR = 159.8987, RMS = 0.2888, CR = 3.2064 (PH), BPS = 3.7425 (PH), and CDR = 112.28 (PH).

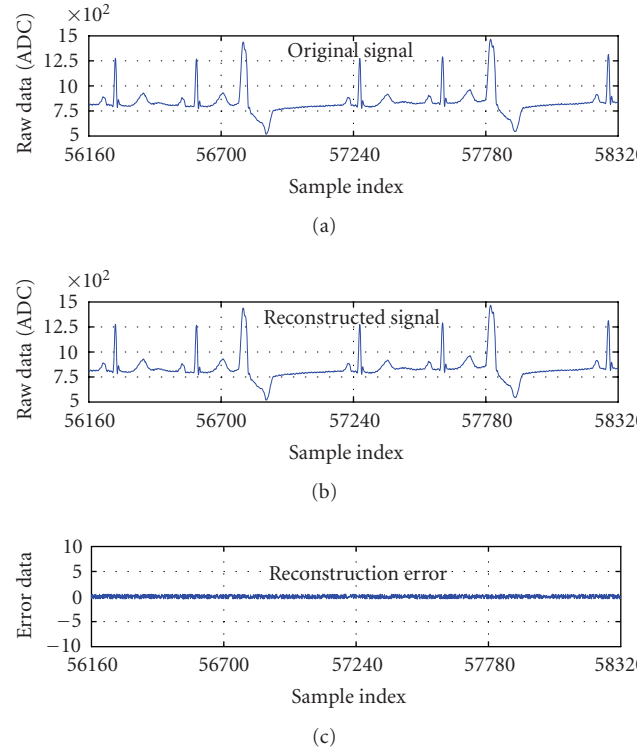

Figure 13: ABT method: original ECG signal along with the reconstructed and residue signals of MIT-BIH ADB record *119MLII*. In the reconstructed signal, PRD = 0.0336, NPRD = 0.2706, SNR = 159.9695, RMS = 0.2888, CR = 3.1507 (PH), BPS = 3.8087 (PH), and CDR = 114.26 (PH).

Table 10: Record MIT-ADB 119: performance comparison results with different ECG compression methods. ASEC denotes analysis by synthesis ECG compressor.

Measure	Proposed scheme (PH)	(DWTC) [11, 12]	(FPWCZ) [33]	(WTDVQ) [15]	(SPIHT) [34]	(TSVD) [31]	(ASEC) [36]	(CAB) [10]	(WPC) [32]
PRD (%)	0.0336	1.88	5.1268	3.3	5.0	6.15	5.48	1.8	19.5
NPRD (%)	0.2706	13.72	—	—	—	—	—	—	—
CR	3.1507	23.0	17.43	—	—	36.72	—	4.0	19.65
CDR	114.2614	—	—	156.5	183	107.7	189.0	—	201.5

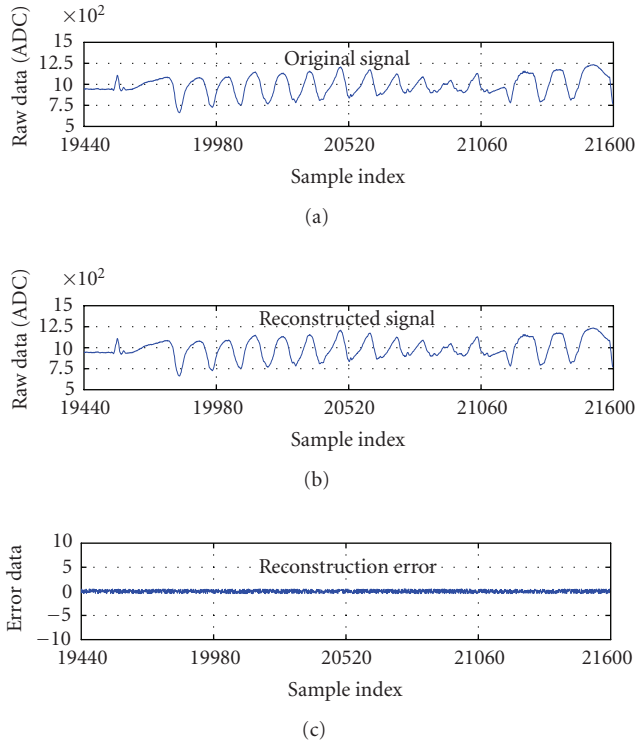

Figure 14: ABT method: original ECG signal along with the reconstructed and residue signals of MIT-BIH ADB record *207MLII*. In the reconstructed signal, PRD = 0.0287, NPRD = 0.3993, SNR = 163.1172, RMS = 0.2879, CR = 3.0583 (PH), BPS = 3.9237 (PH), and CDR = 117.7111 (PH).

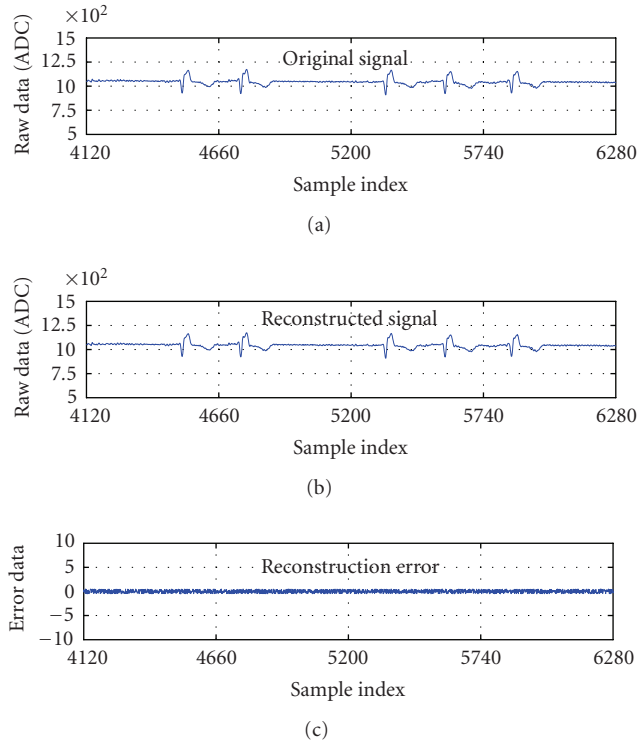

Figure 15: SBT method: original ECG signal along with the reconstructed and residue signals of MIT-BIH ADB record 232V1. In the reconstructed signal, PRD = 0.0281, NPRD = 0.8863, SNR = 163.5516, RMS = 0.2901, CR = 3.4264 (PH), BPS = 3.5023 (PH), and CDR = 105.0681 (PH).

From Figures 16 and 17, it can be noted that NNP-based two-stage compression schemes yield better compression ratio and compressed data rate compared to the LP-based two-stage compression schemes for all the tested records. The improvement in CR values with NNP compared to LP-based compression schemes is determined as follows:

$$\frac{\mathrm{CR}_{\mathrm{NNP}} - \mathrm{CR}_{\mathrm{LP}}}{\mathrm{CR}_{\mathrm{LP}}} \times 100. \tag{22}$$

The values shown in Table 7 are calculated based on the average CR value obtained using all the selected records and it can be observed that the NNP-based schemes result in an improvement of 9.14% and 12.92% on an average compared to the WLSE and MSE algorithms, respectively.

6.2. Comparison with other known methods

Among the proposed single- and two-stage compression schemes, the two-stage compression scheme using the ABT method and Huffman encoder in the second stage (PH) yields the best results. The performance of this scheme is compared with those of other known methods and the results are given in Tables 8–10.

From Tables 8–10, it is observed that the proposed lossless scheme (PH) yields very low PRD (high quality) for all

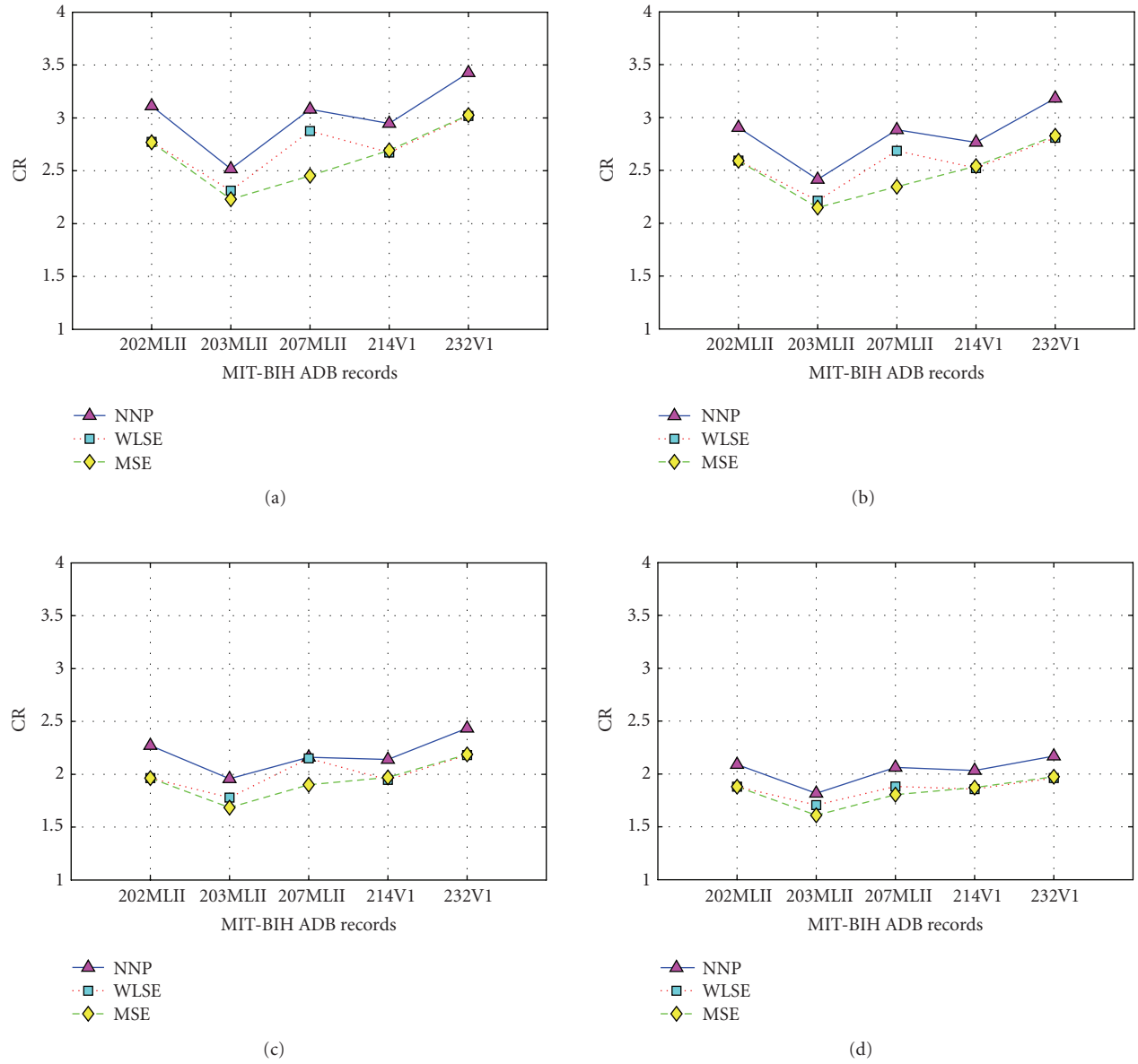

Figure 16: Compression ratio (CR) performance results: (a) PH scheme, (b) PRH scheme, (c) PA scheme, and (d) PRA scheme.

the records even though the compression efficiency is inferior to other lossy methods in the literature. Lossy ECG compression methods usually achieve higher compression ratios compared to lossless methods at low quality. The purpose of this comparison is to examine the tradeoff between compression efficiency and quality for an ECG compression scheme to be used in a particular application. It can be noted that the proposed schemes can be used in applications where the distortion of the reconstructed waveform is intolerable.

7. CONCLUSIONS

This paper has presented lossless compression schemes using multilayer perceptron as a nonlinear predictor in the first stage and different entropy encoders in the second stage. The performances of the schemes have been evaluated using selected records from MIT-BIH arrhythmia database. The experimental results have shown that the compression efficiency of the two-stage method with Huffman coding is nearly twice that of the single-stage method involving only predictor. It is also observed that the proposed ABT and SBT methods yield better compression efficiency performance for long- and short-duration signals, respectively. It is shown that significant improvement in compression efficiency can be achieved with neural network predictors compared to the linear predictors for the same quality with similar setup for different compression schemes. This method yields higher quality of the reconstructed signal compared to other known methods. It can be concluded that the proposed method can

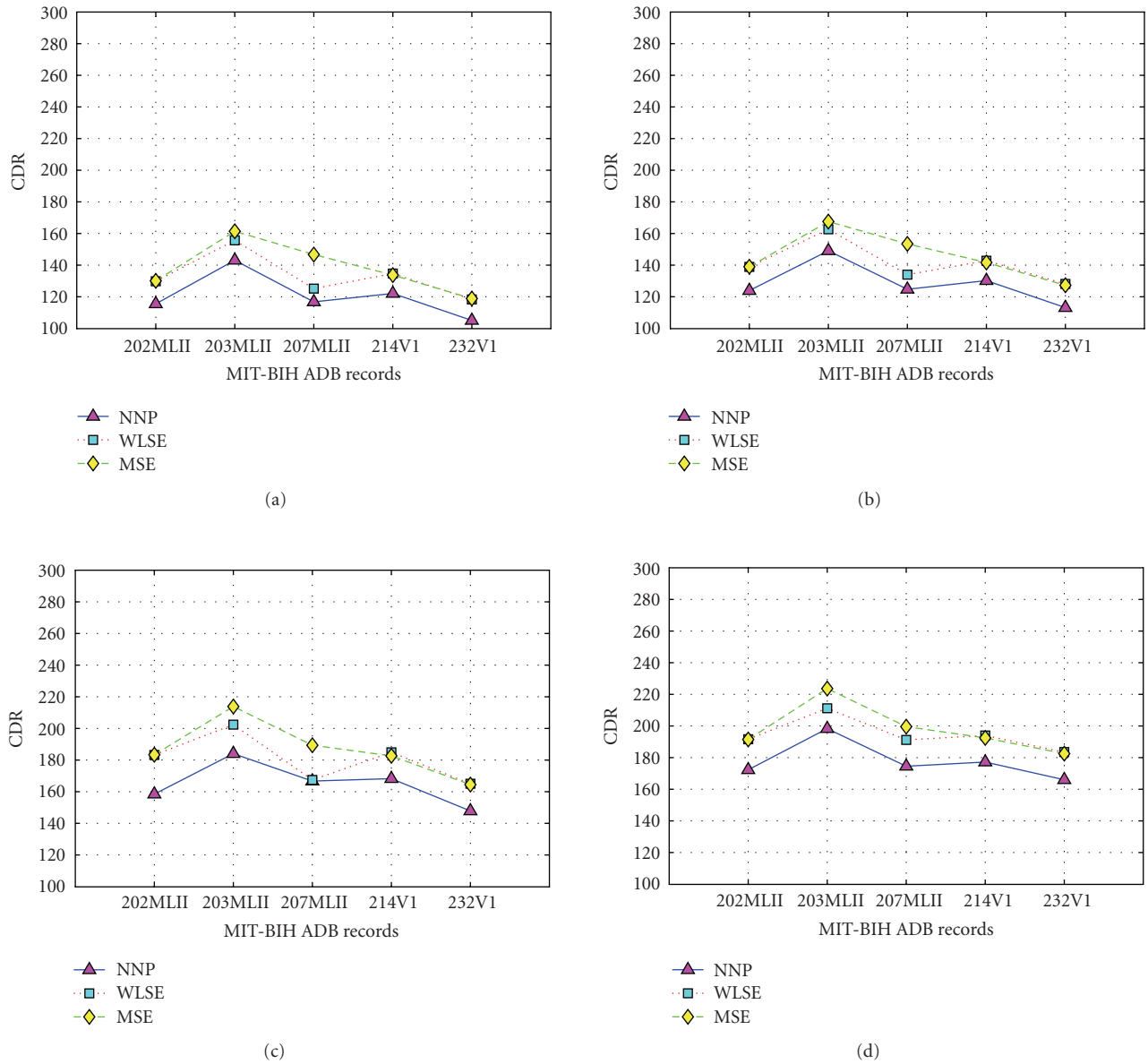

Figure 17: Compressed data rate (CDR) performance results. (a) PH scheme, (b) PRH scheme, (c) PA scheme, and (d) PRA scheme.

be applied to the compression of ECG signals where quality is the main concern compared to the compression efficiency.

ACKNOWLEDGMENTS

This work was supported in part by the IRPA Funding of Government of Malaysia under Grant no. 04-99-01-0052-EA048 and in part by the Internal Funding of Multimedia University under Grant no. PR/2001/0225. We gratefully acknowledge the valuable comments from the reviewers.

REFERENCES

[1] S. M. S. Jalaleddine, C. G. Hutchens, R. D. Strattan, and W. A. Coberly, "ECG data compression techniques—a unified approach," *IEEE Transactions on Biomedical Engineering*, vol. 37, no. 4, pp. 329–343, 1990.

[2] S. D. Stearns, L.-Z. Tan, and N. Magotra, "Lossless compression of waveform data for efficient storage and transmission," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 31, no. 3, pp. 645–654, 1993.

[3] R. D. Dony and S. Haykin, "Neural network approaches to image compression," *Proceedings of the IEEE*, vol. 83, no. 2, pp. 288–303, 1995.

[4] S. A. Rizvi, L.-C. Wang, and N. M. Nasrabadi, "Neural network architectures for vector prediction," *Proceedings of the IEEE*, vol. 84, no. 10, pp. 1513–1528, 1996.

[5] R. Logeswaran and C. Eswaran, "Performance survey of several lossless compression algorithms for telemetry applications," *International Journal of Computers and Applications*, vol. 23, no. 1, pp. 1–9, 2001.

[6] K. Sayood, *Introduction to Data Compression*, Morgan Kaufmann, San Francisco, Calif, USA, 3rd edition, 2006.

[7] M. C. Aydin, A. E. Cetin, and H. Koymen, "ECG data compression by sub-band coding," *Electronics Letters*, vol. 27, no. 4, pp. 359–360, 1991.

[8] S. C. Tai, "Six-band sub-band coder on ECG waveforms," *Medical & Biological Engineering & Computing*, vol. 30, no. 2, pp. 187–192, 1992.

[9] R. S. H. Istepanian, L. J. Hadjileontiadis, and S. M. Panas, "ECG data compression using wavelets and higher order statistics methods," *IEEE Transactions on Information Technology in Biomedicine*, vol. 5, no. 2, pp. 108–115, 2001.

[10] H. Lee and K. M. Buckley, "ECG data compression using cut and align beats approach and 2-D transforms," *IEEE Transactions on Biomedical Engineering*, vol. 46, no. 5, pp. 556–564, 1999.

[11] B. A. Rajoub, "An efficient coding algorithm for the compression of ECG signals using the wavelet transform," *IEEE Transactions on Biomedical Engineering*, vol. 49, no. 4, pp. 355–362, 2002.

[12] A. Alshamali and A. S. Al-Fahoum, "Comments on "An efficient coding algorithm for the compression of ECG signals using the wavelet transform"," *IEEE Transactions on Biomedical Engineering*, vol. 50, no. 8, pp. 1034–1037, 2003.

[13] A. Djohan, T. Q. Nguyen, and W. J. Tompkins, "ECG compression using discrete symmetric wavelet transform," in *Proceedings of the 17th IEEE Annual Conference of Engineering in Medicine and Biology (IEMBS '95)*, vol. 1, pp. 167–168, Montreal, Que, Canada, September 1995.

[14] J. L. Cârdenas-Barrera and J. V. Lorenzo-Ginori, "Mean-shape vector quantizer for ECG signal compression," *IEEE Transactions on Biomedical Engineering*, vol. 46, no. 1, pp. 62–70, 1999.

[15] S.-G. Miaou, H.-L. Yen, and C.-L. Lin, "Wavelet-based ECG compression using dynamic vector quantization with tree codevectors in single codebook," *IEEE Transactions on Biomedical Engineering*, vol. 49, no. 7, pp. 671–680, 2002.

[16] A. Koski, "Lossless ECG encoding," *Computer Methods and Programs in Biomedicine*, vol. 52, no. 1, pp. 23–33, 1997.

[17] C. D. Giurcăneanu, I. Tăbuş, and Ş. Mereuţă, "Using contexts and R-R interval estimation in lossless ECG compression," *Computer Methods and Programs in Biomedicine*, vol. 67, no. 3, pp. 177–186, 2002.

[18] S. D. Stearns, "Arithmetic coding in lossless waveform compression," *IEEE Transactions on Signal Processing*, vol. 43, no. 8, pp. 1874–1879, 1995.

[19] I. H. Witten, R. M. Neal, and J. G. Cleary, "Arithmetic coding for data compression," *Communications of the ACM*, vol. 30, no. 6, pp. 520–540, 1987.

[20] A. Moffat, R. M. Neal, and I. H. Witten, "Arithmetic coding revisited," *ACM Transactions on Information Systems*, vol. 16, no. 3, pp. 256–294, 1998.

[21] G. B. Moody, *MIT-BIH Arrhythmia Database CD-ROM*, Harvard-MIT Division of Health Sciences and Technology, Cambridge, Mass, USA, 3rd edition, 1997.

[22] M. T. Hagan and M. B. Menhaj, "Training feedforward networks with the Marquardt algorithm," *IEEE Transactions on Neural Networks*, vol. 5, no. 6, pp. 989–993, 1994.

[23] M. T. Hagan, H. B. Demuth, and M. Beale, *Neural Network Design*, Thomson Learning, Boston, Mass, USA, 1996.

[24] Y. Zigel, A. Cohen, and A. Katz, "The weighted diagnostic distortion (WDD) measure for ECG signal compression," *IEEE Transactions on Biomedical Engineering*, vol. 47, no. 11, pp. 1422–1430, 2000.

[25] M. Ishijima, "Fundamentals of the decision of optimum factors in the ECG data compression," *IEICE Transactions on Information and Systems*, vol. E76-D, no. 12, pp. 1398–1403, 1993.

[26] A. S. Al-Fahoum, "Quality assessment of ECG compression techniques using a wavelet-based diagnostic measure," *IEEE Transactions on Information Technology in Biomedicine*, vol. 10, no. 1, pp. 182–191, 2006.

[27] C. E. Shannon, "A mathematical theory of communication," *Bell System Technical Journal*, vol. 27, pp. 379–423, 1948.

[28] R. G. Gallagher, "Variations on a theme by Huffman," *IEEE Transactions on Information Theory*, vol. 24, no. 6, pp. 668–674, 1978.

[29] A. H. Sayed, *Fundamentals of Adaptive Filtering*, John Wiley & Sons, New York, NY, USA, 2003.

[30] S. Haykin, *Adaptive Filter Theory*, Prentice-Hall, Upper Saddle River, NJ, USA, 4th edition, 2002.

[31] J.-J. Wei, C.-J. Chang, N.-K. Chou, and G.-J. Jan, "ECG data compression using truncated singular value decomposition," *IEEE Transactions on Information Technology in Biomedicine*, vol. 5, no. 4, pp. 290–299, 2001.

[32] B. Bradie, "Wavelet packet-based compression of single lead ECG," *IEEE Transactions on Biomedical Engineering*, vol. 43, no. 5, pp. 493–501, 1996.

[33] R. Benzid, F. Marir, A. Boussaad, M. Benyoucef, and D. Arar, "Fixed percentage of wavelet coefficients to be zeroed for ECG compression," *Electronics Letters*, vol. 39, no. 11, pp. 830–831, 2003.

[34] Z. Lu, D. Y. Kim, and W. A. Pearlman, "Wavelet compression of ECG signals by the set partitioning in hierarchical trees algorithm," *IEEE Transactions on Biomedical Engineering*, vol. 47, no. 7, pp. 849–856, 2000.

[35] M. L. Hilton, "Wavelet and wavelet packet compression of electrocardiograms," *IEEE Transactions on Biomedical Engineering*, vol. 44, no. 5, pp. 394–402, 1997.

[36] Y. Zigel, A. Cohen, and A. Katz, "ECG signal compression using analysis by synthesis coding," *IEEE Transactions on Biomedical Engineering*, vol. 47, no. 10, pp. 1308–1316, 2000.

R. Kannan was born in Aruppukottai, Tamil Nadu, India, in 1968. He received the B.E. degree in electronics and communication engineering and the Postgraduate Diploma in medical instrumentation technology, both from Coimbatore Institute of Technology, Coimbatore, India, in 1989 and 1990, respectively. He received his M.E. degree in computer science from Regional Engineering College, Trichy, India, in 1995. Since August 2001, he has been working as a Lecturer in the Faculty of Information Technology, Multimedia University, Malaysia. Before that, he has taught both graduate and undergraduate students of Kumaraguru College of Technology, Coimbatore, and other engineering colleges in India for about 10 years. His current research interests include soft computing models and algorithms, biomedical signal processing, time-series forecasting, and data compression. He is a Member of the ISTE, the IEEE Computational Intelligence Society, and the IEEE Engineering in Medicine and Biology Society.

C. Eswaran received his B.Tech., M.Tech., and Ph.D. degrees from the Indian Institute of Technology Madras, India, where he worked as a Professor at the Department of Electrical Engineering till January 2002. Currently, he is working as a Professor in the Faculty of Information Technology, Multimedia University, Malaysia. He has guided more than twenty Ph.D. and M.S. students in the areas of digital signal processing, digital filters, control systems, communications, neural networks, and biomedical engineering. He has published more than 100 research papers in these areas in reputed international journals and conferences. He has carried out several sponsored research projects in the areas of biomedical engineering and communications as Principal Coordinator. He has also served as an Industrial Consultant. He was a Humboldt Fellow in Ruhr University, Bochum, Germany, and was a Visiting Fellow/Faculty in Concordia University, Canada, University of Victoria, Canada, and Nanyang Technological University, Singapore.