A PREDICTIVE APPROACH TO GENERIC ECG DATA COMPRESSION

M. Brito*, J. Henriques*, P. Gil*, M. Antunes**

*CISUC, Centre for Informatics and Systems, University of Coimbra, Coimbra, Portugal
**CCT-HUC, Centre of Cardio-thoracic Surgery, University Hospital of Coimbra, Coimbra, Portugal

{mbrito, jh, pgil}@dei.uc.pt, antunes.cct.huc@sapo.pt

Abstract: In the modern hospital, the efficient storage of electronically recorded biomedical signals as well as its transmission over communication networks is becoming more and more important. Although digital storage media is currently almost inexpensive and computational power has exponentially increased in last years, effective electrocardiogram (ECG) compression techniques are still very attractive. In fact, several millions of electrocardiograms are recorded annually and the transfer of electrocardiogram records over communication networks for remote analysis is now done more than ever. Besides the increased storage capacity for archival purposes, ECG compression allows real-time transmission over communication networks, economic off-line transmission to remote interpretation sites and enables efficient ECG rhythm analysis algorithms.
In this paper, a comparative study is made regarding the suitability of linear and non-linear models for ECG signal compression. More specifically, AR – auto regressive models and Feed-Forward Neural Networks are considered for this task.
The proposed solution provides comparable results with various types of ECG signal, namely normal sinus rhythm, ventricular tachycardia and ventricular fibrillation. Most proposals in the literature approach the problem of ECG compression without considering the possibility of such pathologies and, consequently, their performance deteriorate in cases where such signals are present.
Experimental results are promising, although compression ratios are still not yet as good as they can get.

Introduction

With the growing adoption of clinical information systems and electronic patient records, paper is slowly being abandoned as a means for archiving clinical data. The typical patient file with surgery reports, ECG printouts and vital signal information is now accessible through web-based technologies or specialized applications on most modern hospitals, allowing remote consultation of clinical data. While shifting from a paper-based patient record to a digital one brings many advantages, from the full recording of surgery data to the streaming of data to a portable computing device (such as a PDA) in real-time, the sheer amount of data generated by clinical monitoring equipment introduces the need for compression algorithms.

Among all the areas of medicine, cardiology is one of the branches that require the largest amount of data acquisition for diagnostic purposes. Clinical monitors acquire data from multiple invasive and non-invasive sensors, registering, among others, blood pressure (several channels), ECG, oxygen saturation and temperatures (usually internal and external). Some data channels, such as oxygen saturation and temperature, are usually less critical to represent due to their variation in time. These channels show slow variations and as such need very low sampling intervals (typically, around 1024 *ms* is sufficient). ECG data, however, is particularly critical. In order to be able to accurately represent an ECG signal, clinical systems typically use sampling frequencies ranging from 125 *Hz* to 1 *kHz*, with the higher ones being the most suitable. As an example, data sampled from a set of 12 leads, each generating 12-bit values at 500 Hz results in approximately 30Mb of data per hour [1].

Given that a cardiac surgery can take, on average, around 3 hours, and that a hospital may do around 1000 surgeries per year, a clinical information system will need to handle around 90GB of raw sampled data per year. In the context of dedicated monitoring systems, running 24 hours/day, such as signal monitoring on an intensive care unit or in an arrhythmia-detection system, ECG compression is not only desirable and useful but also necessary.

In recent years, several mobile telemedicine systems design methodologies using different communication technologies were addressed in the literature [2]. Such studies highlighted the need for more efficient compression techniques to overcome the bandwidth limitations of the current generation of cellular telephonic channels for real-time transmission, especially when larger amounts of ECG data are involved. Real-time streaming of vital signals will enable, among other benefits, remote diagnosis and monitoring. Remote diagnosis is important to patients in geographically remote areas, where specialist help is not readily available. Also, transmission from ambulances saves valuable time in emergencies and allows a hospital-based physician to provide better instructions to ambulance personnel. Thus, efficient transmission of biomedical signals over telephone lines or mobile radio is becoming more and more important and compression schemes are a decisive aspect.

One of the less studied aspects in literature is the impact of cardiac anomalies in ECG compression, such as ventricular fibrillation and tachycardia. Under normal circumstances,

ISSN: 1727-1983

the ECG is a very characteristic signal, which can be divided in several waves. The P wave results from initial cardiac excitation. As this excitation reaches the ventricles, a set of three waves is generated, waves Q, R and S, the QRS complex, which is the most striking waveform on an ECG. Once the ventricles have been fully excited they re-polarize, which corresponds to the T wave (Figure 1).

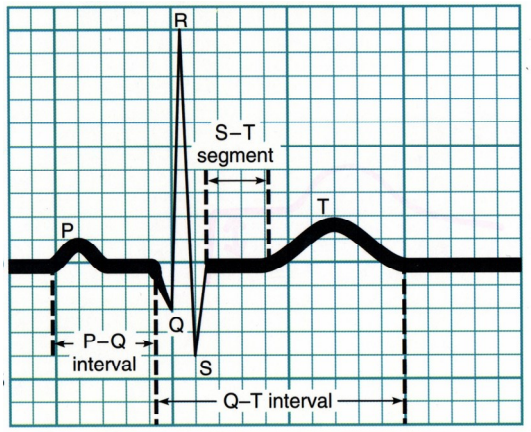

Figure 1: A cardiac cycle in the ECG

Each of these waves provides important clinical information, and possible deviations may be indicative of serious heart pathology. For this reason, it is imperative that the compression algorithm preserve the key characteristics of the ECG signal. Therefore, the purpose of an ECG compression algorithm should be not only to transmit or store the signal with fewer bits, but also to preserve their clinically significant information. An algorithm which provides high compression rates but distorts important features necessary for a correct diagnosis is ultimately useless.

The purpose of this work is to study algorithms based on prediction techniques when applied to anomalous ECG signals, namely ventricular fibrillation and ventricular tachycardia. Linear and non-linear models are used in the prediction stage of the algorithm and an evaluation is made regarding the effectiveness of each approach, taking into consideration computational power, compression ratio and error introduced by the compression process.

Previous Approaches

Digital data compression aims at a space efficient representation of digital data. Conceptually, data compression is the process of detecting and removing redundancies in a given data set [3],[4]. It is possible to distinguish between two kinds of information that are commonly removed: redundant information, which is statistical in nature and can be reconstructed by observing other parts of the signal; and irrelevant information, which is not useful in the target application. Methods that remove redundancies and irrelevancies are often combined. Another important difference between the two classes is that while statistical methods that are used to remove redundancies can be lossless (the method does not result in any data loss; the original signal can be reconstructed from its compressed representation without any error) or lossy (errors are introduced in a controlled manner), the elimination of irrelevancies is intrinsically a lossy process. Lossy techniques are much more effective at compression than lossless methods.

Many data compression techniques for ECG data have been presented in literature [5], [6], [7]. Roughly, they can be classified into two categories: (*i*). Dedicated techniques: these are mainly time domain techniques dedicated to compression of ECG signal. They include the AZTEC, CORTES, Turning Point [5], and FAN [11] algorithms. (*ii*) General techniques: these techniques, which historically were developed for the compression of speech and image/video compression, having a sound mathematical foundation. They include Differential Pulse Code Modulation (DPCM), Subband-and Transform Coding, and Vector Quantization (VQ). There are also approaches described in literature which attempt to modify well known algorithms for the task of ECG compression. Among these, there can be found modifications of the LZ77 [1] and JPEG2000 [12] algorithms specifically for this task. More information about these approaches can be found on the referenced literature. This paper will focus on compression algorithms based on prediction and will evaluate the effectiveness of linear and non-linear models for this task.

The concept of prediction is a general paradigm in data compression. Prediction allows a compact representation of data by encoding the error between the data itself and the information extrapolated or "predicted" from past observations. Such methods exploit redundancy between samples, beats and leads of ECG signal, so that only new information has to be coded. If the predictor works well, predicted samples are similar to the actual input and the prediction error is small or negligible, and so it is easier to encode than the original data.

In a general formulation a source is compressed by predicting the value of each sample from a finite number of past observations. Using this approach, the sample n of signal x can be predicted as function of p previous samples (prediction of order p) by:

$$y(k) = j\,(x(k-1), x(k-2), \ldots, x(k-p)) \qquad (1)$$

where $j(.)$ denotes the predictive function. Boskovic *et al.* [10] have proposed the use of simple linear time invariant systems as prediction model, being the predictive function $j(.)$ is approximated by a linear combination of past samples as

$$x'(k) = \sum_{n=1}^{p} a_n x(k-n) \qquad (2)$$

where $\boldsymbol{a}_1, \boldsymbol{a}_2, \mathbf{K}, \boldsymbol{a}_p$ are the prediction coefficients, optimized to reduce the error variance. These coefficients are evaluated in a way that the average mean square error criterion (3) is minimized:

$$J = \sum_{n=1}^{N} (x(k) - x'(k))^2 \qquad (3)$$

ISSN: 1727-1983

Predictive compression algorithms typically have a quantization step following the predictive stage, where data is discarded and a compromise is made between sample accuracy and compression ratio. The quantized error signal is then encoded losslessly by an entropy encoder (see Figure 2).

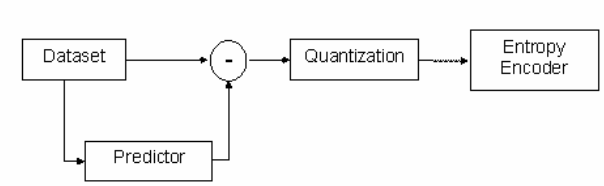

Figure 2: Block diagram of a predictive compression algorithm

In this paper, a comparative study is made between linear and non-linear models, used in the context of a predictive ECG compression algorithm, in particular, AR models and Feed-Forward Neural Networks. The objective is to evaluate the effectiveness of each approach in terms of compression ratio, error introduced by the compression process and tolerance to cardiac anomalies (namely ventricular tachycardia and ventricular fibrillation). Computational complexity is also an important factor, especially in scenarios where the algorithm must be used on devices of reduced computational power (such as a PDA or smart phone).

The proposed approaches achieve compression on the order of between 3:1 and 6:1, compressing the data with acceptable losses in accuracy.

The Proposed Approaches

Linear Predictor

A linear predictor is used to estimate the value of the *kth* sample based on the values of the previous *n* samples. The predictor was built by using an AR model, which was chosen for this task since the entire process can be seen as a system where the outputs of the system are being predicted in function of the previous *n* outputs. Mathematically, it can be described by the following difference equation:

$$y(k) + a_1 y(k-1) + ... + a_{na} y(k-na) = 0 \quad (4)$$

which is a particular case of the ARMA model when the output of the system is solely dependent on the previous outputs of the system:

$$y(k) + a_1 y(k-1) + ... + a_{na} y(k-na) = b_0 u(k) + b_1 u(k-1) + ... + b_{nb}(k-nb) \quad (5)$$

Once the coefficients have been calculated by the least squares method, calculating y(k) is a matter of performing *na* multiplications and additions, which means that to start-up the algorithm it is necessary to store the first *na* samples of the dataset.

The order of the model was determined experimentally, with the objective of achieving a good compromise between the number of parameters of the model and the predictive error. Tests were performed in order to determine the impact of the model order on the fitness of the model to the data in question. Since the model coefficients are calculated adaptively for each frame, the matter of over-fitting the model is not the problem it would be on other types of adaptive systems; instead, the model that provides the least predictive error would be preferable. However, it was determined that there was no significant gain from using a model order larger than 4. In fact, the overhead introduced by storing the model coefficients is higher than the gain obtained from a slightly lower prediction error. For this reason, Akaike's Information Criterion (AIC) and MDL were not used as criterion for adaptive model order selection, since it always selects the highest order model (Figure 3).

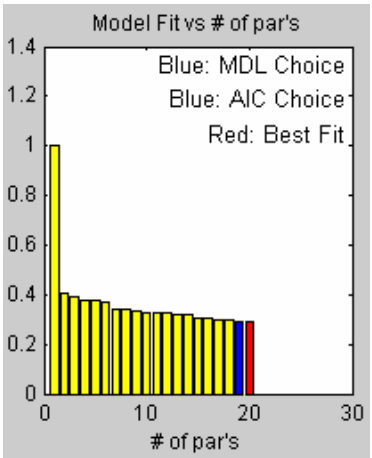

Figure 3: Model fit vs model order (order 1 through order 20)

The algorithm works in frames of configurable size and the coefficients are calculated for each frame specifically. The most adequate size for each frame would depend on what would be considered an acceptable delay in ECG signal streaming. In the next section, it will be explained that the overhead of starting a new frame is considerable, so the choice of a frame size must be based on a compromise between transmission delay and compression ratio.

The first step of the algorithm is to perform normalization of the ECG data to integer values ranging between 0 and 1024. After this is done, the predictor calculates the value of the next sample by considering the *n* previous data samples. Compression is achieved by storing the error between the predicted values and the real ECG samples. If the predictor is good enough, the error signal will be low in amplitude and the quantization process will return a relatively reduced number of symbols which can be efficiently encoded with an entropy encoder.

ISSN: 1727-1983

In the proposed algorithm, all calculations in the prediction phase are performed in floating point arithmetic. The errors resulting from the prediction phase are then quantized and rounded to integers before being stored by an entropy decoder. For now, a simple Huffman [8] encoder is being used. The error signal is being currently quantized by using a fixed factor.

Computationally, the algorithm is very simple and should be viable in scenarios where limited computational power is available. The decompression process is extremely lightweight, producing a new sample with only *n* multiplications and additions. The compressor, although more computationally intensive than the decompressor, should be relatively lightweight as well, since the process to determine the model coefficients is a least squares method, which involves a matrix transposition, multiplication and inversion.

Non-Linear Predictor

For the following ECG compression experiments, a Feed-Forward Neural Network was chosen for the prediction stage of the algorithm. A multi-layered network is being used for this task with eight neurons on the hidden layer. There are four neurons on the input layer, which means that the neural network will consider the previous four outputs of the system when calculating the predicted output. There is only one neuron on the output layer, the predicted value for sample *k*. The network uses sigmoid activation functions on the input and hidden layer, and a linear function on the output layer.

The network is trained at start-up time for a maximum of 150 epochs using the Levenberg-Marquardt algorithm, with an error goal of 10E-1, stopping the training process when one of these conditions is reached. Calculating a new model for each frame has prohibitive costs, which would impact both the compression ratio and the ability of the algorithm to work on a real-time scenario. Therefore, the objective is to train the neural network well enough for a given frame, while still allowing it to maintain generalization capabilities which will provide a good predicted signal on frames where similar ECG signals are present.

Re-training the neural network periodically is a possibility, but one that must be avoided at all costs. The cost of storing 40 weights periodically will introduce a severe penalty in the resulting compression ratio, which will be particularly important as frame sizes get smaller. This has special importance in the context of real-time data streaming in a telemedicine application, where the time spent re-training the neural network may introduce unacceptable delays in data transmission. Since the predictive error may increase due to the transition between typical and atypical ECG signals, such as a tachycardia or fibrillation, this work will also study the impact of re-training the neural network on the overall compression ratios and signal reconstruction error.

With the exception of the modifications in the predictor stage of the algorithm, the subsequent stages remain otherwise unchanged, in order to get consistent results. The same quantization and encoding processes for compressing the predictive error signal were used in both linear and non-linear versions of the algorithm and will not be described again.

Experimental Results

The proposed algorithm was tested on a variety of signals, from normal ECGs to sustained fibrillation, in order to study the compression rate obtained for each of the cases, as well as the error introduced by the compression process. The same signals were applied to both versions of the algorithm, in order to get an accurate measurement of how each predictor influences the resulting compression ratios and signal degradation.

Raw ECG data is being read from a binary file, with each sample coded in 16-bit. In order to measure the error introduced by the algorithm, the *percentage root mean difference* (PRD) is used. This criterion for distortion measurement is very common in literature and allows us to get an idea of the error introduced by the compression process, independently of the size of the frame being used by the algorithm.

$$PRD = \sqrt{\frac{\sum_{i=1}^{N}(y(i)-y'(i))^2}{\sum_{i=1}^{N}y(i)^2}} \quad (6)$$

It should be noted, however, that the PRD is not a perfect form of measurement of the ECG degradation because some regions of the ECG waveform are much more important than others for clinical diagnosis. However, it has been consistently used in literature as a signal degradation measurement, so it should not be ignored.

The amount of compression achieved by the algorithm is defined by the ratio between the number of bits necessary to describe the data originally and the number of bits necessary to describe the compressed data. The inverse ratio may be used, in order to obtain a percentage relative to the original size instead of an *n-to-1* ratio. (7)

$$\boldsymbol{CR} = \frac{\boldsymbol{Compressed\ length\ (bits)}}{\boldsymbol{Original\ length\ (bits)}} \quad (7)$$

Algorithm Validation

In order to get consistent results, both variants of the algorithm were tested with frame sizes between 256 and 2048 samples, duplicating the frame size each time. For validation purposes, each algorithm was applied to ECG signals downloaded from the MIT-BIH database. Each signal being considered contains periods of normal and abnormal ECG activity. This is an important point, since it will allow us to observe the efficiency of the algorithms in each of these scenarios. For brevity, only signal *vfdb/419*, from the MIT-

ISSN: 1727-1983

BIH Malignant Ventricular Ectopy Database, is presented in this paper.

In the algorithm using the linear AR model, tests were made in order to determine the impact of the predictor order and frame size in the compression rate and signal error. Predictor orders 2 through 5 were explored in the performed tests. The dataset was compressed using a quantization factor of 6, which resulted in an overall PRD of approximately 0.7%.

Table 1 summarizes the resulting compression ratios for several frame sizes and model orders. For these tests, a subset containing the first 50000 samples of the dataset was compressed.

Table 1: Compression rates for various frame sizes and predictor orders (AR predictor)

	2	3	4	5
256	3.9:1	3.98:1	3.86:1	3.75:1
512	4.63:1	4.76:1	4.68:1	4.59:1
1024	5.1:1	5.31:1	5.24:1	5.19:1
2048	5.4:1	5.64:1	5.61:1	5.59:1

Although the algorithm does not present any performance degradation on areas of ventricular fibrillation, the overhead of storing the AR model along with each frame introduces an overhead which becomes more noticeable as the frame sizes get smaller and when the model order increases.

For this reason, there is a change in strategy when using a neural network. Since the number of coefficients that must be stored is much higher (close to 40), the algorithm trains a new neural network for the first frame being encoded and uses the same network as the predictor for the following frames.

Table 2: Compression rates for various frame sizes (Neural Network)

Compression rate	256	512	1024	2048
Observations	3.89:1	4.64:1	5.26:1	5.5:1
	4.21:1	4.72:1	4.86:1	5.59:1
	3.39:1	4.47:1	5.18:1	5.29:1
	3.86:1	4.8:1	5.14:1	5.45:1
Mean	3.83:1	4.65:1	5.11:1	5.45:1

Table 2 summarizes the resulting compression ratios for several frame sizes. It must be noted that due to the training of the neural network, each time the network is trained a different solution is reached. Therefore, there are slight variations in the compression ratios reached for each frame size. Since this happens, several observations will be presented, as well as the average compression ratio. The neural network has four inputs, so its results may be compared to the case where the AR model has order 4.

Re-training the neural network when reaching a given threshold did not provide any improvements on compression ratio. In fact, it reduced the compression ratio substantially due to the overhead of storing the network weights.

The following figures (Figure 4, Figure 5 and Figure 6) provide graphical representations of the samples in the dataset, before and after being subjected to each of the algorithms. Visual inspection reveals that there is little signal degradation, noticeable mostly at higher zoom levels.

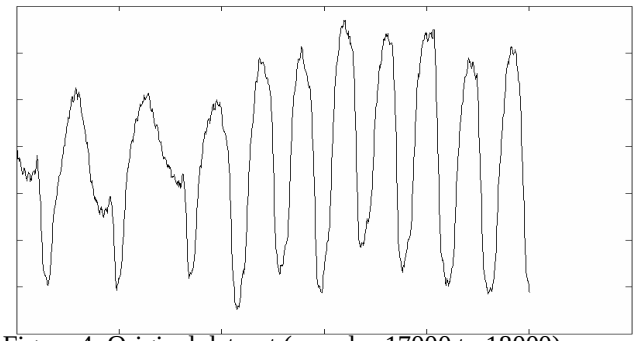

Figure 4: Original dataset (samples 17000 to 18000)

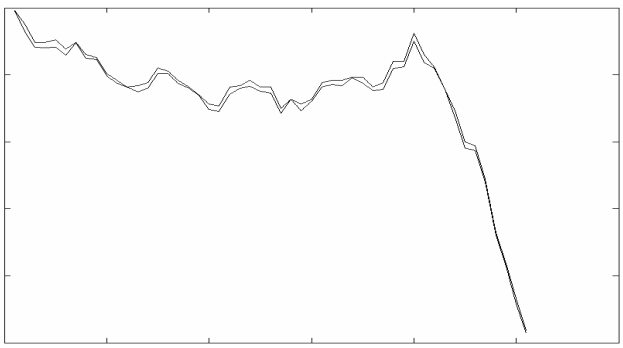

Figure 5: AR Predictor Algorithm: Original vs Reconstructed dataset (zoom at samples 17000 to 17050)

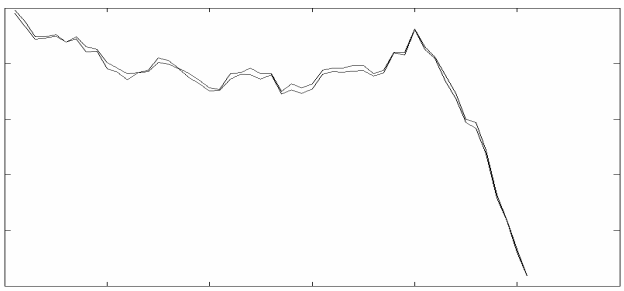

Figure 6: Neural Network Predictor Algorithm: Original vs Reconstructed dataset (zoom at samples 17000 to 17050)

From these experiments, it can be concluded that there is no clear advantage in using a neural network as a predictor for ECG signal compression. Although other topologies will be explored in the future, it is preferable to use a linear AR model, since it involves simpler calculations both when

ISSN: 1727-1983

compressing and decompressing the data. Although processing power is getting cheaper each day, this is an especially critical factor in scenarios of reduced computational power, such as a PDA or similar mobile devices.

Conclusions

From the performed studies we can draw several conclusions. The size of the frame is necessarily related the amount of buffering that a streaming application needs to make in order to decode and present the data. A large frame may introduce a shift of several seconds between the data that is being compressed and the data that can be presented to the user at any given time. If a large delay is not acceptable, smaller frame sizes should be selected. However, the smaller a frame gets, the lower the compression ratio will be, due to overhead introduced by initializing a new frame. More specifically, since the number of possible error values resulting from quantization remains approximately the same, there is a fixed penalty from storing the Huffman dictionary in both cases. In the solution using a linear AR model, there is also the need to store a fixed number of model coefficients for each frame, which also results in a higher overhead as frame sizes get smaller.

Error quantization is the factor that needs most improvement in this algorithm. A fixed quantizer is currently being used in the algorithm. However, examination of the reconstructed signal reveals that a there are areas of the ECG where the quantization process introduces some noticeable oscillations. Determining a quantization factor adaptively could provide better results, both in terms of accuracy and compression ratio. The quantization can be determined either through the calculation of a quantization factor or by introducing a new model that would provide more information regarding the error signal.

Experimental results indicate that the proposed approaches work well in encoding anomalous ECG signals, such as ventricular tachycardia. The algorithms seem to be tolerant to deviations from the normal sinus rate without noticeable performance degradation.

Conclusions and Future Work

The presented algorithms, although providing moderate compression ratios (between 3:1 and 6:1, depending on the parameters being used), successfully compresses ECG signals of several types with little loss of precision.

Predictive algorithms using linear AR models and neural networks were explored in this paper. It can be concluded from experimental results that no significant advantage exists from using a neural network, since an AR model provides comparable results at a more reduced computational cost.

The work presented on this paper concentrated on the predictive stage of the algorithm, which means that the subsequent stages need improvement. The proposed approach to ECG compression is clearly not a finished work; in fact it is just getting started, so there are many aspects where it can be improved. One issue that has much room for improvement is the error quantization step. At present, error quantization is being done by a fixed factor, but it's easy to understand that this is inefficient and introduces slight irregularities on some areas of the ECG signal. Therefore, if the error quantization is done adaptively in each frame, it would be possible to minimize the irregularities introduced in the reconstructed signal while maximizing the improvements in compression ratio provided by this step of the algorithm.

Huffman codification of the quantized error signal seems to be efficient; however an attempt should be made to find a more efficient variable-length code for usage on this scenario. The overhead introduced by the codification of the Huffman dictionary results in a severe hit in the compression ratio which may be possible to avoid by using entropy encoders which do not require a symbol dictionary.

The predictor being used in the algorithm should clearly be topic of further study. The parameters of the predictor can be calculated in a way which will favour smooth error variations, by a regularization technique, for example. The resulting predictive error will hopefully be composed mainly by low frequencies, which can be encoded after applying a Fourier or Cosine transform on the signal, followed by an entropy encoder. This improvement could do much to improve the compression ratio, since the predicted signal presents many oscillations.

References

[1] HORSPOOL R. and WARREN J. (1995): 'An LZ Approach to ECG Compression', Dept. of Computer Science, University of Victoria, Canada.

[2] ISTEPANIAN R. H., BALOS P., WOODWARD B., CHEN S., and LUK B.: 'The comparative performance of mobile telemedical systems using the IS-54 and GSM standards', J. Telemedicine Telecare, vol. 5, no. 2, pp. 97–104.

[3] SHANON C.E., WEAVER W. (1949): 'The Mathematical Theory of Communication', University Illinois Press, Urbane, IL.

[4] SAYOOD, K. (1996): 'Introduction to Data Compression', Morgan Kaufmann Publisher, San Francisco, pp.181-206.

[5] JALALEDDINE S.M., HUTCHENS C.G., STRATTAN R.D., CORBERLY W.A. (1990), 'ECG data compression techniques: a unified approach', IEEE Trans. Biomed. Eng. **37** pp. 329-343.

[6] SHRIDHAR M., MOHANKRISHNAN N. (1984), 'Data compression techniques for electrocardiograms', Electric Eng. J. 9 (4), pp. 126-131.

[7] MARI J. (1997), 'Compresion de ECG en Tiempo Real con el DSP TMS320C25', Univeritat de Valencia, Tesis de Licenciatura, February.

[8] HUFFMAN D. (1951), 'A method for the Construction of Minimum Redundancy Codes'.

[9] WELCH T. (1984) 'A technique for High-Performance Data Compression', June 1984.

[10] BOSKOVIC A., DESPOTOVIC M., BAJIAE D. (2004), 'Predictive ECG coding using linear time-invariant models', Arch Oncology 2004;12(3):152-8.

[11] DIPERSIO D. A. and BARR R.C., (1985) 'Evaluation of the fan method of adaptive sampling on human electrocardiograms', Medical & Biological Engineering & Computing, pp. 401- 410, September 1985.

[12] BILGIN A., MARCELLIN M., ALTBACH M. (2003), 'Compression of Electrocardiogram Signals using JPEG2000', IEEE Transactions on Consumer Electronics, Vol. 49, No. 4, November 2003.

ISSN: 1727-1983