ELSEVIER

Signal Processing 59 (1997) 43–59

SIGNAL PROCESSING

ECG data compression by spline approximation

Marta Karczewicz[a], Moncef Gabbouj[b,*]

[a]*Nokia Research Center, P.O. Box 100, Sinitaival 6, FIN-33721 Tampere, Finland*

[b]*Signal Processing Laboratory, Tampere University of Technology, P.O. Box 553, FIN-33101 Tampere, Finland*

Received 14 December 1995; revised 31 January 1997

Abstract

A novel and efficient transform method for ECG data compression based on B-spline basis functions is proposed. The algorithm allows these basis functions to adapt their shape to the nonstationary behavior of ECG signals. The number and shape of these basis functions are completely characterized by the number and location of the so-called knots. The position of the knots can effectively be coded using run-length coding. Therefore, the overhead data required to describe the way the B-spline basis functions vary in time can be neglected compared to the gain in compression achieved by adaptation. The quasi-periodic nature of the ECG signal is used in order to further reduce redundant information in the data. Low bit-rates of the order of 160–200 bits/s are achieved with very good quality of the reconstructed signal. The algorithm is compared with other transform-based schemes (DCT and DLT), and was found superior at any bit-rate. © 1997 Elsevier Science B.V.

Zusammenfassung

Es wird eine neue und effiziente Methode, die ECG Daten mit Hilfe von Spline-Basisfunktionen komprimiert, vorgeschlagen. Der Algorithmus ist in der Lage die Basisfunktionen dem nichtstationären Verhalten der ECG Signale anzupassen. Die Anzahl und die Form der Basisfunktionen sind komplett durch die Anzahl und die Position ihrer sogenannten Knoten bestimmt. Die Position der Knoten kann effektiv in eine "run-length"-Kodierung transformiert werden. Die zusätzlichen Daten, mit denen die Zeitvarianz der Basisfunktionen beschrieben wird, kann man deshalb gegenüber dem Kompressionsgewinn durch die Adaption vernachlässigen. Aufgrund des quasiperiodischen Verhaltens von ECG Signalen kann die Redundanz der Kodierung weiter reduziert werden. Geringe Datenraten von 160–200 bits/s werden erreicht, wobei die Qualität des rekonstruierten Signals sehr gut ist. Im Vergleich mit anderen Transformationsverfahren (DCT und DLT) erzielt dieser Algorithmus bei jeder Bitrate bessere Ergebnisse. © 1997 Elsevier Science B.V.

Résumé

Une nouvelle méthode efficace pour la compression d'ECG basée sur les fonctions B-splines est proposée. L'algorithme permet l'adaptation des fonctions de base au caractère non stationnaire des signaux ECG. Le nombre et la forme de ces fonctions de base sont complètement caractérisées par le nombre et la position des noeuds. La position des noeuds peut être codée de manière efficace en utilisant un codage des répétitions. De cette façon, on peut négliger l'importance de l'entête du fichier des données décrivant la manière dont les B-splines varies dans le temps en comparaison avec le gain de compression atteint par cette adaptation. La nature quasi périodique de l'ECG est utilisée pour réduire la redondance dans les données. Des bas débits de l'ordre de 160-200 bits/s ont été attents avec une trés bonne reconstruction du signal original. L'algorithme

* Corresponding author. Fax: +358 3 316 1857; e-mail: moncef@cs.tut.fi.

PII S0165-1684(97)00037-6

est comparé avec d'autres techniques basées également sur des transformations (DCT et DLT), et s'est avéré meilleur pour n'importe quel débit. © 1997 Elsevier Science B.V.

Keywords: ECG data compression; Adaptive compression; Spline functions

1. Introduction

Electrocardiogram (ECG) signals are usually sampled at 200–500 samples/s with 8–12 bits resolution. Considering long monitoring periods, compression is required to handle such vast amount of data. It can increase the capacity of databases where hundreds of thousands of ECG signals are stored for subsequent monitoring and evaluation. Other applications of ECG compression include transmission via telephone or mobile radio to an ECG center for further processing.

In recent years many algorithms for ECG data compression have been suggested. For a recent review of these methods the reader is referred to [10]. Existing methods for data compression can be classified into three categories: (1) direct data handling methods, (2) transformation-based methods, and (3) parameterized model-based methods [23]. Our approach belongs to the second group. In general, transform techniques involve expanding a signal into a weighted sum of basis functions. The coefficients of this sum are properly encoded and stored or transmitted instead of the original data. The best transform is the one which requires the least number of basis functions to represent the input signal for a given mean-square error (mse). The signal-dependent Karhunen–Loeve transform (KLT) is the optimal orthogonal transform in this sense. It has been employed for ECG data compression in [1]. The basis functions of KLT are eigenvectors of the covariance matrix of the original data. Adapting the KLT to a single signal requires sending or storing all of its basis functions, point by point, because these functions have no simple structure and cannot be coded efficiently. The overhead involved in coding them can far outweigh the possible gain in compression. In practice, the KLT is optimized in the mean performance for all signals in the given database, and there is just one set of KLT basis functions which is used for the whole class of signals.

The KLT is often replaced by input-independent transforms such as discrete cosine transform (DCT) [1], fast Walsh transform (WT) [15] or discrete Legendre transform (DLT) [21]. On the one hand, the DCT is the most widely used in signal compression because it closely approximates the KLT especially for data modeled as first-order autoregressive signals. However, it works well only under the assumption that the signal is wide-sense stationary and consequently its application to ECG compression produced poor results. DLT, on the other hand, requires half the number of bits per sample to represent an R–R interval with the same reconstruction error compared to DCT [21]. The reason is that the local properties of the DCT basis functions do not vary in time, while the local amplitude and frequency of DLT functions are changing in time in a way that better resembles the nonstationary behavior of ECG signal in an R-R interval. The performance of transform-based coding methods strongly depends on the shape of basis functions. Optimally, the functions should adapt to the nonstationary behavior of ECG signals, and the information describing the way they vary in time should require as small number of bits per sample as possible.

Splines play an important role in approximation theory. They have been applied to model 1-D and 2-D data in [12–14, 25, 26]. In this paper, B-splines are proposed as a set of adaptive basis functions to be used for ECG signal compression. B-spline-based transform requires a much lower bit-rate than fixed transforms as it can adapt to the nonstationary behavior of an ECG signal. The information describing the way in which the basis functions vary in time can be coded very efficiently. The number and shape of these functions are completely characterized by the number and location of the so-called knots and their local properties can be changed by varying the position of the knots. By reducing the number of knots, the number of basis functions is reduced. The position of the knots can be effectively coded using run-length coding. Philips proposed another adaptive transform,

namely time-warped polynomial transform [20]. Unlike time-warped polynomials, B-splines are well localized in time and are not oscillatory by nature. These properties can be advantageous when modeling certain types of signals.

Previous work using spline approximations for ECG compression [9, 16, 24] fixed the location of knots according to a set of rules involving the local values of amplitude and curvature of the signal. An additional refinement of knot positions during which the location of knots is varied, one at a time, was developed in [24]. In this paper, a novel and efficient solution for the problem of approximating data by splines for which not only the coefficients but also the position and number of the knots are optimized is presented. The idea behind the proposed reduction method is to start with an initial spline approximation given by a large number of knots. The knots are then removed, one by one, without perturbing the initial spline approximation by more than a given tolerance. The number and location of these knots are determined automatically.

The paper is organized as follows. In Section 2, a review of the B-spline model is presented. A general description of the proposed data reduction approach is given in Section 3. In Section 4, an ECG compression method based on the developed strategy for knot removal is presented. Simulation results and performance comparison are provided in Section 5. Section 6 contains some conclusions.

2. Splines and B-splines

A spline function consists of polynomial pieces on subintervals joined together by continuity conditions. The segmented nature allows splines to adjust very efficiently to the local characteristics of the data and represent it better (i.e. with smaller deviations) than other classes of functions.

Definition 2.1. A function $f(x)$, defined on a finite interval $[a, b]$, is called a spline function of order $k > 0$, having as knots the sequence $\boldsymbol{t} = \{t_0, t_1, \ldots, t_{n+1}\}$ ($t_0 = a, t_{n+1} = b$) such that $t_i < t_{i+k}$ (the knots coordinates t_i may not be distinct), if the following two conditions are satisfied:

1. In each knot interval $[t_i, t_{i+1}]$, $f(x)$ is given by a polynomial of degree $k - 1$ at most.
2. At any knot t_i such that $t_{i-1} < t_i = \cdots = t_{i+l} < t_{i+l+1}$, the function $f(x)$ has continuous $k - l - 2$ derivatives (and is discontinuous at t_i if $l = k - 1$).

The vector space of functions satisfying Definition 2.1 will be denoted by $S_{k,t}$. The dimension of the vector space $S_{k,t}$ is

$$\dim(S_{k,t}) = n + k. \tag{1}$$

To perform computations with splines, one must choose a suitable representation in which any member of $S_{k,t}$ can be written as a unique linear combination of $n + k$ basis functions such that Definition 2.1 is automatically satisfied. A common choice is to use B-splines. Computations with B-splines are particularly convenient, due to their local-support property, i.e. they are nonzero only over a finite interval. Moreover, B-splines are a unique minimum-support basis – the only set of basis functions in which each covers the minimum number of knots. Using B-splines, curve-fitting problems are easy to pose and lead to well-conditioned, banded positive-definite systems. They also provide an easy to manipulate representation for splines having different degrees of smoothness at each knot.

Definition 2.2. A B-spline $B_{i,k,t}(x)$ of order $k > 0$, with knots $t_i, \ldots, t_{i+k}$, can be defined using the following recurrence relation:

$$B_{i,1,t}(x) = \begin{cases} 1 & \text{if } t_i \leqslant x < t_{i+1}, \\ 0 & \text{otherwise,} \end{cases} \tag{2}$$

and

$$B_{i,r,t}(x) = \frac{t_{i+r} - x}{t_{i+r} - t_{i+1}} B_{i+1,r-1,t}(x) + \frac{x - t_i}{t_{i+r-1} - t_i} B_{i,r-1,t}(x) \tag{3}$$

for $r = 2, 3, \ldots, k$, where we interpret the terms

$$\frac{t_{i+r} - x}{t_{i+r} - t_{i+1}} \quad \text{and} \quad \frac{x - t_i}{t_{i+r-1} - t_i} \tag{4}$$

as zero whenever $t_{i+r} - t_{i+1} = 0$ and $t_{i+r-1} - t_i = 0$, respectively.

From this definition one can observe that B-splines have local support

$$B_{i,k,t}(x) = 0 \quad \text{if } x \notin [t_i, t_{i+k}]. \tag{5}$$

With the given set of knots $t_i, i=0,\dots,n+1$, one can construct $n-k+2$ linearly independent B-splines of order k, i.e. $B_{i,k,t}(x)$ for $i=0,\dots,n-k+1$. To obtain a full set of basis functions for the vector space $S_{k,t}$, one needs another $2k-2$ linearly independent splines. This can be accomplished by introducing additional knots $t_{-k+1},\dots,t_{-1}$ and $t_{n+2},\dots,t_{n+k}$ satisfying

$$t_{-k+1} \leqslant t_{-k+2} \leqslant \cdots \leqslant t_{-1} \leqslant t_0 = a, \tag{6}$$

$$b = t_{n+1} \leqslant t_{n+2} \leqslant \cdots \leqslant t_{n+k-1} \leqslant t_{n+k}, \tag{7}$$

but which are otherwise arbitrary. Every function $f(x)$ satisfying Definition 2.1 then has a unique representation (*the Curry–Schoenberg Theorem*)

$$f(x) = \sum_{i=-k+1}^{n} a_i B_{i,k,t}(x), \tag{8}$$

in which a_i is called the ith B-spline coefficient of $f(x)$. The upper bound for the absolute value of the spline function $f(x)$ is given by [3]

$$\max_x |f(x)| \leqslant \max_i |a_i|. \tag{9}$$

The most common choice for boundary knots are coincident knots

$$t_{-k+1} = t_{-k+2} = \cdots = t_{-1} = t_0 = a, \tag{10}$$

$$b = t_{n+1} = t_{n+2} = \cdots = t_{n+k-1} = t_{n+k}. \tag{11}$$

This choice implies that all B-splines vanish outside the interval $[a,b]$ and allows very easily to impose the boundary conditions

$$f(a) = a_{-k+1}, \qquad f(b) = a_n. \tag{12}$$

The knots $t_1,\dots,t_n$ are called interior knots, and in the data reduction scheme, these are the only knots which can be eliminated by the knot removal algorithm.

One of the most powerful tools in studying B-splines also used extensively in this algorithm is an explicit formula which allows us to write basis functions $B_{i,k,\rho}(x)$, defined for the knot sequence $\rho = t \backslash \{t_p\}$, $t = \{t_{-k+1},\dots,t_{n+k}\}$, in terms of basis functions $B_{i,k,t}(x)$ defined for t [2]:

$$B_{i,k,\rho}(x) = \begin{cases} B_{i,k,t}(x) & \text{if } i \leqslant p-k-1, \\ \dfrac{t_p - t_i}{t_{i+k} - t_i} B_{i,k,t}(x) + \dfrac{t_{i+k+1} - t_p}{t_{i+k+1} - t_{i+1}} B_{i+1,k,t}(x) & \\ & \text{if } p-k \leqslant i \leqslant p-1, \\ B_{i+1,k,t}(x) & \text{if } p \leqslant i \leqslant n-1. \end{cases} \tag{13}$$

3. Compression by knot removal

The compression problem can be stated as follows. Given a given set of data values y_i corresponding to data indices x_i, $i=1,\dots,N$, and a class of functions $g(x;\theta)$ parametrized by θ, find the function parameter θ which for the chosen norm $\|\cdot\|$ minimizes

$$\|y - g(\theta)\|, \tag{14}$$

where

$$y = [y_1,\dots,y_N]^{\mathrm{T}}, \tag{15}$$

$$g(\theta) = [g(x_1;\theta),\dots,g(x_N;\theta)]^{\mathrm{T}}. \tag{16}$$

The best approximating function is the one which can be described by the smallest number of parameters when the error measured by (14) is below a given threshold. The least-squares norm is usually used as the closeness criterion

$$\frac{1}{N}\sum_{i=1}^{N}(y_i - g(x_i;\theta))^2. \tag{17}$$

Approximation based on this norm has the following advantages:

1. Minimization of the least-squares norm has easy to implement, not iterative solutions.
2. Approximation based on least-squares norm is an unbiased estimate for the signal corrupted by zero mean, white noise with a finite variance. Moreover, if the measurement errors have a Gaussian distribution, and if the measured data are considered to be a realization of a stochastic process, then the obtained estimate is the minimum variance estimate among all unbiased estimators.

The spline represented as a linear combination of basis B-spline functions (8) has been chosen as the approximating function. Therefore, the parameters to be determined are:

1. the order k of the spline,
2. the number and the location of knots,
3. the coefficients a_i.

Experimentally, it was found that for the application at hand, the use of cubic splines ($k=4$) gives a good trade-off between efficiency (computation time and memory requirement) and the quality of fit. If the order of the spline and the knots are fixed, the least-squares problem (17) becomes a linear one since the spline function is linear in the unknown B-spline coefficients, as follows from (8). The coefficients can be easily calculated as the solution to the overdetermined linear system of equations. However, the number and shape of the B-spline basis functions depend on the number and position of the knots (2), (3) which, in this case, must be fixed a priori. The quality of the approximation thus depends on the proper choice of knots. A least-squares problem with the number and position of knots as variable parameters must consequently be considered. Since splines depend non-linearly on the knots as can be seen from Eqs. (2) and (3), the criterion to be minimized is non-linear. Moreover, it may have several local minima and it is almost impossible to decide when and if a global minimum has been reached [11].

In classical methods, after the initial choice of knots is selected their location is varied in order to determine the position which minimizes the least-squares error. In the simplest algorithms, knot locations are changed one by one, starting with the rightmost interior knot and proceeding sequentially to the left [4]. This method was used to compress low-frequency segments of ECG data [24]. The initial knots are chosen in the locations exhibiting large changes in the data values (changes in the slope), such as peak or valley points. The knots can also be relocated simultaneously using a non-linear multivariate minimization algorithm, such as the conjugate gradient method of Fletcher and Reeves. In both cases, the resulting solution depends strongly on the initial position of the knots. Since the function to be minimized is non-convex [11], one can only expect to find a local minimum in the vicinity of the initial location. In both approaches, the number of knots which must be known a priori is kept constant during optimization. If the number of knots is too small, the corresponding fit is not accurate enough; if it is too large, redundant information is still left in the data. Therefore, an additional technique must be developed to determine the number of knots. The other problem is computational complexity. In both methods, each function evaluation after changing the position of knots requires the computation of a fixed-knot, least-squares splines which involves the solution of the overdetermined system of equations.

In this paper, a new and efficient approach for determining the number and position of knots for a given data is introduced. Let $f(x)$, defined on the knot sequence $\boldsymbol{t}=\{t_{-k+1},\ldots,t_{n+k}\}$, be a spline interpolation to a given data ($f(x_j)=y_j$, $j=1,\ldots,N$):

$$f(x)=\sum_{i=-k+1}^{n} a_i B_{i,k,\boldsymbol{t}}(x). \tag{18}$$

The number of basis functions must be equal to the number of data points $n+k=N$ to ensure interpolation [6]. The problem considered here is to determine the subspace $S_{k,\tau}$ of $S_{k,t}$ of the lowest possible dimension (the shortest knot sequence τ) and an element $g(x)$ of $S_{k,\tau}$ such that

$$\|\boldsymbol{f}-\boldsymbol{g}\|\leqslant\varepsilon, \tag{19}$$

where

$$\boldsymbol{f}=[f(x_1),\ldots,f(x_N)]^{\mathrm{T}}, \tag{20}$$

$$\boldsymbol{g}=[g(x_1),\ldots,g(x_N)]^{\mathrm{T}}, \tag{21}$$

and ε is a given non-negative real number. Denote by $\boldsymbol{\tau}^l$ a subsequence obtained by removing l knots from $\boldsymbol{t}$ and by $g_l(x)$ an approximation to $f(x)$ defined on $\boldsymbol{\tau}^l$. The proposed strategy involves three main steps:

1. Initialize: $g_0(x)=f(x)$ and $\boldsymbol{\tau}^0=\boldsymbol{t}$;
2. For $l=1,2,\ldots$
   (a) decide which knot should be removed from $\boldsymbol{\tau}^{l-1}$ and create a new subsequence $\boldsymbol{\tau}^l$;
   (b) compute the approximation g_l defined on $\boldsymbol{\tau}^l$ to the given data.

The iterations are stopped when

$$\|\boldsymbol{f}-\boldsymbol{g}_l\|>\varepsilon, \tag{22}$$

and the approximation $g_{l-1}(x)$ is the solution. The new spline approximation at each iteration is computed to

minimize the least-squares criterion (17). The error as measured by the least-squares norm does not give any information about the maximum difference between the spline $f(x)$ and its approximation $g_l(x)$ at a specific point. For this reason, the ∞-norm is selected as the termination criterion (22):

$$\max_x |f(x) - g_l(x)| > \varepsilon_\infty. \tag{23}$$

Although the ∞-norm has the disadvantage of not being robust against noise (a single noise spike may lead to a large norm), its use ensures the preservation of diagnostically significant features of the ECG waveform.

Removing knots, one at a time, allows one to avoid problems related to interaction between neighboring knots and to find a computationally efficient algorithm for determining the importance of each knot in representing the underlying data. Moreover, it will be shown that the coefficients of the new spline approximation g_l can be obtained at very low computational cost by updating the solution from the previous iteration g_{l-1}.

A B-spline compression algorithm based on iterative knot removal is presented in [18]. There are substantial differences between these two algorithms. In [18] the knots are removed in groups which makes it more difficult to determine the significance of each knot in representing the spline due to the dependency between neighboring knots. Unlike in the proposed method, the knot ranking algorithm in [18] requires solving a system of equations. The spline fit to the initial data is re-computed at each iteration using the discrete approximation method suggested in [17]. For the spline of order $k = 4$ the achieved root mean square error can be 10 times larger than the one obtained from exact solution [17].

3.1. Least-squares approximation with splines

The ECG signal is divided into intervals. In each of these, a spline approximation is used. To further improve compression ratio and avoid discontinuities, the first and the last point in each interval are interpolated, not approximated. Consider a given set of data values y_j corresponding to data indices x_j, $j = 1,\ldots,N$, with $x_1 = a$, $x_N = b$. At the lth iteration of the knot removal algorithm, one needs to find the coefficients of spline $g_l(x)$ of order k with given knots $\tau^l = \{\tau^l_{-k+1},\ldots,\tau^l_{m+k}\}$,

$$g_l(x) = \sum_{i=-k+1}^{m} c_i^l B_{i,k,\tau^l}(x), \tag{24}$$

where $m = n - l$, such that

$$\sum_{j=2}^{N-1} (y_j - g_l(x_j))^2 \tag{25}$$

is minimized subject to

$$g_l(x_1) = y_1, \qquad g_l(x_N) = y_N. \tag{26}$$

When choosing coincident boundary knots (10), (11) and using the boundary condition in Eq. (12), constraint (26) results in

$$c^l_{-k+1} = y_1, \qquad c^l_m = y_N. \tag{27}$$

Instead of minimizing (25), the following expression can now be minimized:

$$\sum_{j=2}^{N-1} \left(\bar{y}_j - \sum_{i=-k+2}^{m-1} c_i^l B_{i,k,\tau^l}(x_j) \right)^2, \tag{28}$$

where

$$\bar{y}_j = y_j - c^l_{-k+1} B_{-k+1,k,\tau^l} - c^l_m B_{m,k,\tau^l}, \quad j = 2,\ldots,N-1. \tag{29}$$

Therefore, the problem reduces to the determination of the B-spline coefficients c_i^l, $i = -k+2,\ldots,m-1$, as the solution, in the least-squares sense, of the overdetermined linear system

$$\sum_{i=-k+2}^{m-1} c_i^l B_{i,k,\tau^l}(x_j) = \bar{y}_j, \quad j = 2,\ldots,N-1. \tag{30}$$

To avoid recomputing the whole spline approximation after removing a knot at each iteration, a QR factorization-based algorithm will be developed and used to update the solution. This algorithm can be much more efficient when the approximated data $\bar{y}_j$ in Eq. (30) remain constant during iterations. The approximated data given by Eq. (29) depend on the basis functions which are determined by the sequence of knots and therefore can be modified as a result of

knot removal. To avoid this problem, the original data is replaced by

$$\tilde{y}_j = y_j - \frac{y_N - y_1}{x_N - x_1}x_j - \frac{x_N y_1 - x_1 y_N}{x_N - x_1}, \quad j = 1, \ldots, N, \tag{31}$$

prior to iterations. As a result

$$\tilde{y}_1 = 0 \quad \text{and} \quad \tilde{y}_N = 0. \tag{32}$$

When employing $\tilde{y}_j$ instead of the original data in Eq. (27) one has

$$c^l_{-k+1} = 0, \qquad c^l_m = 0, \tag{33}$$

Substituting these values of c^l_{-k+1} and c^l_m into (29) yields

$$\tilde{y}_j = \tilde{y}_j, \quad j = 2, \ldots, N-1. \tag{34}$$

Thus, the approximated data remain constant during knot removal. Eq. (30) can be written in a matrix notation as

$$\boldsymbol{E}_l \boldsymbol{c}_l = \tilde{\boldsymbol{y}}, \tag{35}$$

where

$$\boldsymbol{E}_l = \begin{bmatrix} B_{-k+2,k,\tau^l}(x_2) & \cdots & B_{m-1,k,\tau^l}(x_2) \\ \vdots & & \vdots \\ B_{-k+2,k,\tau^l}(x_{N-1}) & \cdots & B_{m-1,k,\tau^l}(x_{N-1}) \end{bmatrix}, \tag{36}$$

$$\tilde{\boldsymbol{y}} = \begin{bmatrix} \tilde{y}_2 \\ \vdots \\ \tilde{y}_{N-1} \end{bmatrix}, \qquad \boldsymbol{c}_l = \begin{bmatrix} c^l_{-k+2} \\ \vdots \\ c^l_{m-1} \end{bmatrix}. \tag{37}$$

To determine the least-squares solution of system of equations (35), the following orthogonalization method is used [8]. Assume that we have the QR factorization of the matrix $\boldsymbol{E}_l$,

$$\boldsymbol{E}_l = \boldsymbol{Q}_l \boldsymbol{R}_l = \boldsymbol{Q}_l \begin{bmatrix} \boldsymbol{R}_{l1} \\ \boldsymbol{0} \end{bmatrix}, \tag{38}$$

where $\boldsymbol{Q}_l$ is an orthonormal matrix and $\boldsymbol{R}_{l1}$ a square upper triangular matrix of order $m+k-2$. Furthermore, let

$$\boldsymbol{z}_l = \begin{bmatrix} \boldsymbol{z}_{l1} \\ \boldsymbol{z}_{l2} \end{bmatrix} = \boldsymbol{Q}_l^{\mathrm{T}} \tilde{\boldsymbol{y}}, \tag{39}$$

with $\boldsymbol{z}_{l1}$ having $m + k - 2$ elements. The vector of spline coefficients $\boldsymbol{c}_l$ is the solution of the triangular system

$$\boldsymbol{R}_{l1} \boldsymbol{c}_l = \boldsymbol{z}_{l1}, \tag{40}$$

which can be obtained by simple backsubstitution.

To find a new prediction error at $(l+1)$th iteration, after removing one knot, one does not have to evaluate matrices $\boldsymbol{R}_{l+1}$ and $\boldsymbol{z}_{l+1}$ from the beginning, which would be very computationally complex considering the size of matrix $\boldsymbol{E}_{l+1}$. The QR factorization of matrix $\boldsymbol{E}_{l+1}$ can be very efficiently computed by updating the factorization of $\boldsymbol{E}_l$.

The matrix $\boldsymbol{E}_{l+1}$ consisting of the new basis functions $B_{i,k,\tau^{l+1}}(x)$ defined for the knot sequence $\boldsymbol{\tau}^{l+1} = \boldsymbol{\tau}^l \backslash \{\tau^l_p\}$ can be obtained from the matrix $\boldsymbol{E}_l$ using Eq. (13). In matrix notation

$$\boldsymbol{E}_{l+1} = \boldsymbol{E}_l \boldsymbol{B}, \tag{41}$$

where

$$\boldsymbol{B} = \begin{bmatrix} \boldsymbol{I}_{p-2} & 0 & 0 \\ 0 & \boldsymbol{B}_{22} & 0 \\ 0 & 0 & \boldsymbol{I}_{m-p-1} \end{bmatrix}, \tag{42}$$

$\boldsymbol{I}_{p-2}$, $\boldsymbol{I}_{m-p-1}$ are identity matrices of order $p-2$ and $m-p-1$, respectively, and

$$\boldsymbol{B}_{22} = \begin{bmatrix} \lambda_{p-1} & \cdots & 0 \\ \mu_{p-1} & \ddots & \vdots \\ \vdots & \ddots & \lambda_{p+k-2} \\ 0 & \cdots & \mu_{p+k-2} \end{bmatrix}, \tag{43}$$

with

$$\lambda_i = \frac{\tau^l_p - \tau^l_{i-k+1}}{\tau^l_{i+1} - \tau^l_{i-k+1}} \tag{44}$$

and

$$\mu_i = \frac{\tau^l_{i+2} - \tau^l_p}{\tau^l_{i+2} - \tau^l_{i-k+2}}. \tag{45}$$

From Eqs. (38) and (41), one has

$$\boldsymbol{E}_{l+1} = \boldsymbol{E}_l \boldsymbol{B} = \boldsymbol{Q}_l \boldsymbol{R}_l \boldsymbol{B} = \boldsymbol{Q}_l \tilde{\boldsymbol{R}}. \tag{46}$$

To obtain $\boldsymbol{R}_{l+1}$ and $\boldsymbol{Q}_{l+1}$, matrix $\tilde{\boldsymbol{R}}$ should be triangularized. Note that the columns of $\tilde{\boldsymbol{R}}$ are either

unchanged columns of the upper triangular matrix $\boldsymbol{R}_l$ or linear combinations of two consecutive columns of this matrix. Thus, matrix $\tilde{\boldsymbol{R}}$ is upper Hessenberg and the unwanted subdiagonal elements $\tilde{r}_{p,p-1},\ldots,\tilde{r}_{m+k-2,m+k-3}$ can be zeroed by a sequence of Givens rotations

$$\boldsymbol{G}_{m+k-3}^{\mathrm{T}}\cdots\boldsymbol{G}_{p-1}^{\mathrm{T}}\tilde{\boldsymbol{R}}=\boldsymbol{R}_{l+1}, \tag{47}$$

where $\boldsymbol{G}_i^{\mathrm{T}}$ transforms the ith and $(i+1)$th row vectors

$$\begin{bmatrix} 0 & \cdots & 0 & r_{i,j} & r_{i,j+1} & r_{i,j+2} & \cdots \\ 0 & \cdots & 0 & r_{i+1,j} & r_{i+1,j+1} & r_{i+1,j+2} & \cdots \end{bmatrix} \tag{48}$$

into

$$\begin{bmatrix} 0 & \cdots & 0 & \tilde{r}_{i,j} & \tilde{r}_{i,j+1} & \tilde{r}_{i,j+2} & \cdots \\ 0 & \cdots & 0 & 0 & \tilde{r}_{i+1,j+1} & \tilde{r}_{i+1,j+2} & \cdots \end{bmatrix} \tag{49}$$

through the following formulas:

$$\tilde{r}_{i,j}=\sqrt{r_{i,j}^2+r_{i+1,j}^2},$$
$$c=r_{i,j}/\tilde{r}_{i,j},$$
$$s=r_{i+1,j}/\tilde{r}_{i,j},$$
$$\tilde{r}_{i,l}=cr_{i,l}+sr_{i+1,l}\quad l=j+1,\ldots,$$
$$\tilde{r}_{i+1,l}=-sr_{i,l}+cr_{i+1,l}\quad l=j+1,\ldots$$

The matrix $\boldsymbol{E}_l$ has the band structure with at most k non-zero adjacent elements in a row. As a consequence, $\boldsymbol{R}_l$ and $\tilde{\boldsymbol{R}}$ have also the band structure with k adjacent non-zero elements [5]. This further reduces the computational complexity because the Givens rotation can now be restricted to the vectors of length $k+1$. Matrix $\boldsymbol{Q}_l$ does not need to be updated. The vector $\boldsymbol{z}_{l+1}$ can be obtained by applying Givens transformations to the corresponding elements of $\boldsymbol{z}_l$ ($\boldsymbol{z}_l$ can be considered as an extra column of $\tilde{\boldsymbol{R}}$) [5].

3.2. Ranking of the knots

Next, we describe the algorithm which allows us to estimate the significance of each interior knot in representing the spline function g_l given by

$$g_l(x)=\sum_{i=-k+1}^{m} c_i^l B_{i,k,\tau^l}(x), \tag{50}$$

and decide which knot should be removed. Suppose that knot $z=\tau_p^l$ is to be removed. The function $h(x)$ defined on $\rho=\rho_{-k+1},\ldots,\rho_{m+k-1}=\tau^l\backslash\{z\}$ is given by

$$h(x)=\sum_{i=-k+1}^{m-1} d_i B_{i,k,\rho}(x). \tag{51}$$

A natural approach is to let the mean-square error between g_l and h,

$$\frac{1}{N}\sum_{j=1}^{N}(h(x_j)-g_l(x_j))^2 \tag{52}$$

be the measure of significance of the knot τ_p^l in representing spline g_l. To find values of d_i, $i=-k+1,\ldots,m-1$ for which expression (52) is minimized, an overdetermined linear system of N equations should be solved. However, only a rough estimate of the relative importance of the knots is needed. Thus, to avoid extensive computations, an approximate solution to the problem of minimizing Eq. (52) is used. First, h is represented using the same basis B_{i,k,τ^l} as function g_l [2]:

$$h(x)=\sum_{i=-k+1}^{m} \tilde{d}_i B_{i,k,\tau^l}(x), \tag{53}$$

where

$$\tilde{d}_i=\begin{cases} d_i & \text{if } i\leqslant p-k,\\ \alpha_i d_i+(1-\alpha_i)d_{i-1} & \text{if } p-k+1\leqslant i\leqslant p-1,\\ d_{i-1} & \text{if } p\leqslant i\leqslant m, \end{cases} \tag{54}$$

$$\alpha_i=\frac{z-\rho_i}{\rho_{i+k-1}-\rho_i}. \tag{55}$$

Now, the mse between g_l and h is given by

$$\frac{1}{N}\sum_{j=1}^{N}\left(\sum_{i=-k+1}^{m}(c_i^l-\tilde{d}_i)B_{i,k,\tau^l}(x_j)\right)^2. \tag{56}$$

Coefficients d_i, $i=-k+1,\ldots,m-1$, are found, such that $c_i^l=\tilde{d}_i$ for all i except $i=i_0$ for some i_0. From Eq. (54), one immediately has

$$d_i=\begin{cases} c_i^l & \text{if } i\leqslant p-k,\\ c_{i+1}^l & \text{if } p-1\leqslant i\leqslant m-1. \end{cases} \tag{57}$$

For the remaining coefficients, there are two different possible expressions stemming from (54):

$$d_i = \frac{c_i^l - (1-\alpha_i)d_{i-1}}{\alpha_i},$$

$$i = p-k+1,\ldots,p-2, \tag{58}$$

or

$$d_{i-1} = \frac{c_i^l - \alpha_i d_i}{1-\alpha_i},$$

$$i = p-1, p-2,\ldots,p-k+2. \tag{59}$$

Substituting (57) and (58) into (54), we have $\tilde{d}_i = c_i^l$ for all i but $i = p-1$. If (59) is used instead of (58), then $\tilde{d}_i = c_i^l$ for all i but $i = p-k+1$. The mean square error for these two alternatives is, respectively, given by

$$\varepsilon_p^1 = \frac{1}{N}\sum_{j=1}^{N}((c_{p-1}^l - \tilde{d}_{p-1})B_{p-1,k,\tau^l}(x_j))^2 \tag{60}$$

and

$$\varepsilon_p^2 = \frac{1}{N}\sum_{j=1}^{N}((c_{p-k+1}^l - \tilde{d}_{p-k+1})B_{p-k+1,k,\tau^l}(x_j))^2. \tag{61}$$

The weight

$$w_p = \min(\varepsilon_p^1, \varepsilon_p^2) \tag{62}$$

is assigned to the knot τ_p^l. This weight is a measure of the significance of this knot in representing the spline g_l.

If a function g_l is itself an approximation to a spline f, the weight becomes

$$w_p = \min(\zeta_p^1, \zeta_p^2), \tag{63}$$

where

$$\zeta_p^1 = \frac{1}{N}\sum_{j=1}^{N}(f(x_j) - h(x_j))^2$$

$$= \frac{1}{N}\sum_{j=1}^{N}((f(x_j) - g_l(x_j)) + (g_l(x_j) - h(x_j)))^2$$

$$= \frac{1}{N}\sum_{j=1}^{N}((f(x_j) - g_l(x_j)) + (c_{p-1}^l - \tilde{d}_{p-1})B_{p-1,k,\tau^l}(x_j))^2, \tag{64}$$

$$\zeta_p^2 = \frac{1}{N}\sum_{j=1}^{N}(f(x_j) - h(x_j))^2$$

$$= \frac{1}{N}\sum_{j=1}^{N}((f(x_j) - g_l(x_j)) + (g_l(x_j) - h(x_j)))^2$$

$$= \frac{1}{N}\sum_{j=1}^{N}((f(x_j) - g_l(x_j)) + (c_{p-k+1}^l - \tilde{d}_{p-k+1})B_{p-k+1,k,\tau^l}(x_j))^2. \tag{65}$$

This procedure is repeated for all interior knots τ_p^l, $p = 1,\ldots,m$. The knot with the smallest weight w_p is removed from $\boldsymbol{\tau}^l$ and the new subsequence is created $\boldsymbol{\tau}^{l+1} = \boldsymbol{\tau}^l \backslash \{\tau_p^l\}$. The new spline approximation to the original $g_{l+1}(x)$ defined on this subsequence is found. If the resulting error is smaller than the predefined threshold the iterations are continued. Otherwise, $\boldsymbol{\tau}^l$ becomes the final knot sequence. The computation of ε_p^1 and ε_p^2 can be accelerated using local support property (5) and considering only $x_j \in [\tau_{p-1}, \tau_{p+k-1}]$ for B_{p-1,k,τ^l} in Eqs. (60) and (64) and $x_j \in [\tau_{p-k+1}, \tau_{p+1}]$ for B_{p-k+1,k,τ^l} in Eqs. (61) and (65).

The influence of using the approximate solution to the minimization problem (52) on the decision which knot should be removed is established experimentally. For 100 different intervals, the final spline approximation and number of knots is found by two versions of the knot removal algorithm. In one, criterion (52) is minimized by solving the overdetermined system of equation in the least-squares sense; while, in the other, the proposed solution is utilized. Both versions are applied with the same stopping criterion. The average number of the remaining basis functions was equal to 22 when employing the exact solution and to 24 using the approximate one.

4. ECG compression

ECG compression algorithm presented here uses the fact that the ECG is a quasi-periodic signal with

strong correlation between adjacent beats. The detection of P and T waves is not simple and sometimes not even feasible; thus, the R–R interval is selected to be the repetitive period. The ECG signal is segmented into R–R intervals using the method presented in [7]. The ECG is passed through a differentiator and low-pass filter. The output signal is thresholded using two thresholds, equal in magnitude but opposite in polarity. The number and the distance between alternate threshold crossing is used to detect R waves.

For each interval, the knot sequence t_{prev} from the previous interval is used to compute an approximation of the knot sequence for the current data, t_{curr}, whose ith entry, t_i^{curr} is given by

$$t_i^{\text{curr}} = t_i^{\text{prev}} \frac{N_{\text{curr}}}{N_{\text{prev}}}, \tag{66}$$

where N_{curr} and N_{prev} are the lengths of the current and previous intervals, respectively. Note that the current interval is reinitialized to start at 0. Only when the mse is larger than some threshold ε_{mse} or the absolute difference between the compressed and original data is larger than ε_∞ of peak-to-peak signal value, we search for a new knot sequence using the algorithm described in Section 3. This reduces not only the computation time but also the amount of overhead data needed to update basis functions. The threshold ε_{mse} strongly depends on the noise level. Thus, a proper choice for ε_{mse} may be $\varepsilon_{\text{mse}} = \alpha\varepsilon_{\text{prev}}$, where $\varepsilon_{\text{prev}}$ is the mse computed for the interval for which the knot sequence was originally found.

The results can be further improved by building a codebook of some limited number of previously coded R–R complexes and corresponding knot sequences and coefficients. The code of the member has to be transmitted or stored so that the proper knot sequence is used at the decoder to reconstruct the signal. For the current R–R complex, one of the knot sequences found in the codebook, resulting in the smallest mse, should be chosen. This requires computing spline approximation using all the codebook knot sequences. Another approach is to find in the codebook an R–R complex which is closest to the current one in some sense and use its knot sequence. The selection of the codebook member can be done by any pattern-matching method such as maximum correlation or minimum distance. Experimentally, it was found that this does not significantly deteriorate the performance of the algorithm. A possible choice, which was determined experimentally, for ε_{med} is $\varepsilon_{\text{mse}} = \alpha\varepsilon_{\text{med}}$, where ε_{med} is the median of mse values for the intervals in the codebook. To use the correlation between adjacent beats as described above, it is assumed that the number of knots remains constant during the compression for most of the intervals. Thus, when searching for the new knot sequence the iterations are stopped when

(1) absolute difference between the compressed and original data is larger than ε_∞ of peak-to-peak signal value, or
(2) the number of basis functions (coefficients) n is equal to n_{const}.

If $n = n_{\text{const}}$ the new knot sequence replaces the oldest one in codebook.

The knots are coded as a difference between the position of two consecutive knots using Huffman coding. For n coefficients one has to send n knots. Because the value of the first knot is always set to zero, $n-1$ differences have to be coded. The coefficients c_i of the spline function are quantized, using a uniform quantization scheme. The quantization step Δ is the same for all the coefficients in all intervals, but it adapts to the dynamic range of the signal. This is done by taking Δ as a fraction β of the peak-to-peak amplitude of the signal. The influence of coefficients quantization on the reconstructed signal can be easily analyzed. Let c_i be the original and $\hat{c}_i$ the quantized B-spline coefficients, and let $g(x_j)$ and $\hat{g}(x_j)$, $j = 1,\ldots,N$, be the function values for unquantized and quantized coefficients, respectively. Using expression (9) one has

$$\begin{aligned}\max_j |\hat{g}(x_j) - g(x_j)| &= \max_j \left| \sum_{i=-k+1}^{m} (\hat{c}_i - c_i) B_{i,k,\tau}(x_j) \right| \\ &\leqslant \max_i |\hat{c}_i - c_i|, \end{aligned} \tag{67}$$

and for the uniform quantizer

$$\max_i |\hat{c}_i - c_i| < \frac{\Delta}{2}. \tag{68}$$

The quantized coefficients are stored for the current R–R interval whenever a new knot sequence must be found. On the other hand, when a close match is found for the current R–R interval in the codebook (containing previously stored R–R intervals, their knot

Table 1
Data structure of the compressed R–R interval

Contents	Number of bits
Value of the first data point in the interval	11
Length of the codeword used to represent the coefficient (the difference of coefficients) in the interval	4
B-spline coefficients	Varying
Information about knot sequence	
Usage of codebook	1
With codebook	
– Position in codebook	3
Without codebook	
– Number of knots $n = n_{\text{const}}$ $(n > n_{\text{const}})$	1 (6)
– Position of knots	Varying

sequences and coefficients) only the differences between the quantized coefficients (those of the current and the matching intervals) are stored, instead. In either case, a constant number of bits per coefficient (or difference) is used and, for each interval, the maximum number of bits needed to store the coefficients (or the differences) is saved. Note that the differences between the coefficients in two intervals are only stored when the same number of knots is used for both intervals. The parameters required to describe one interval and the bit allocation per parameter is given in Table 1.

The constants ε_∞, α and β have been determined empirically and set to be $\varepsilon_\infty = 2.5\%$, $\alpha = 2$ and $\beta = 0.01$. Since the value of peak-to-peak amplitude of the signal can be changed significantly by the base line shift, the base line is removed using the notch filter presented in [22].

5. Results

The proposed method was tested using 22 records from the MIT-BIH Arrhythmia database [19] from the first leads (360 Hz with 11 bits/sample resolution). Each record was 1 min in duration.

The amount of compression is characterized by the number of bits required to represent the compressed signal (coefficients and overhead information, such as interval length and position of knots) per second (bps). As an objective measure of "goodness", we use the percentage root-mean-square difference (PRD), calculated as

$$\text{PRD} = 100\sqrt{\frac{\sum_i (\hat{y}_i - y_i)^2}{\sum_i y_i^2}}, \tag{69}$$

where y_i and $\hat{y}_i$ are the ith original and reconstructed sample, respectively.

The proposed algorithm is compared using the same database with discrete Legendre transform DLT [21]. To ensure fair comparison the coefficients of the Legendre transform are quantized and stored in the same way as the spline coefficients in the proposed method. The quantized coefficients or the difference between the coefficients obtained for the current interval and one of the intervals in the codebook can be stored depending which option results in a smaller number of bits. The codebook is updated when for the interval under consideration storing directly the quantized coefficients is selected. The current R–R interval replaces the oldest interval in the codebook. To make the comparison between the DLT and the proposed method easier, the number of basis functions (coefficients) n per R–R interval is set to be a fixed value for both methods during the compression. Both methods are applied with different number of coefficients ($n = 20, 25, 30, 35, 40, 45, 50$) yielding various compression ratios and values of error. The results are summarized in Fig. 1. The mean value of PRD error for the proposed scheme is lower than that of DLT at any bit-rate. Note that to achieve the same PRD as spline approximation of order, e.g. $n = 20$, DLT requires 40 basis functions. To indicate the number of bits used for updating the shape of basis functions, bit-rates calculated without and with the extra data for the knots sequences (BR_1 and BR_2, respectively) are given in Table 2. The proposed approach is also compared against DCT [1]. However, even with 50 DCT basis functions, the PRD for the same database is still high, 19.9, with bit-rate 306.

The PRD values can be misleading especially for noisy signals. Noise reduction will produce large PRD errors. The PRD error alone does not reveal whether an algorithm can preserve diagnostically important features of the ECG waveforms. In many situations, diagnostically important small waveforms can be distorted or eliminated without significant changes

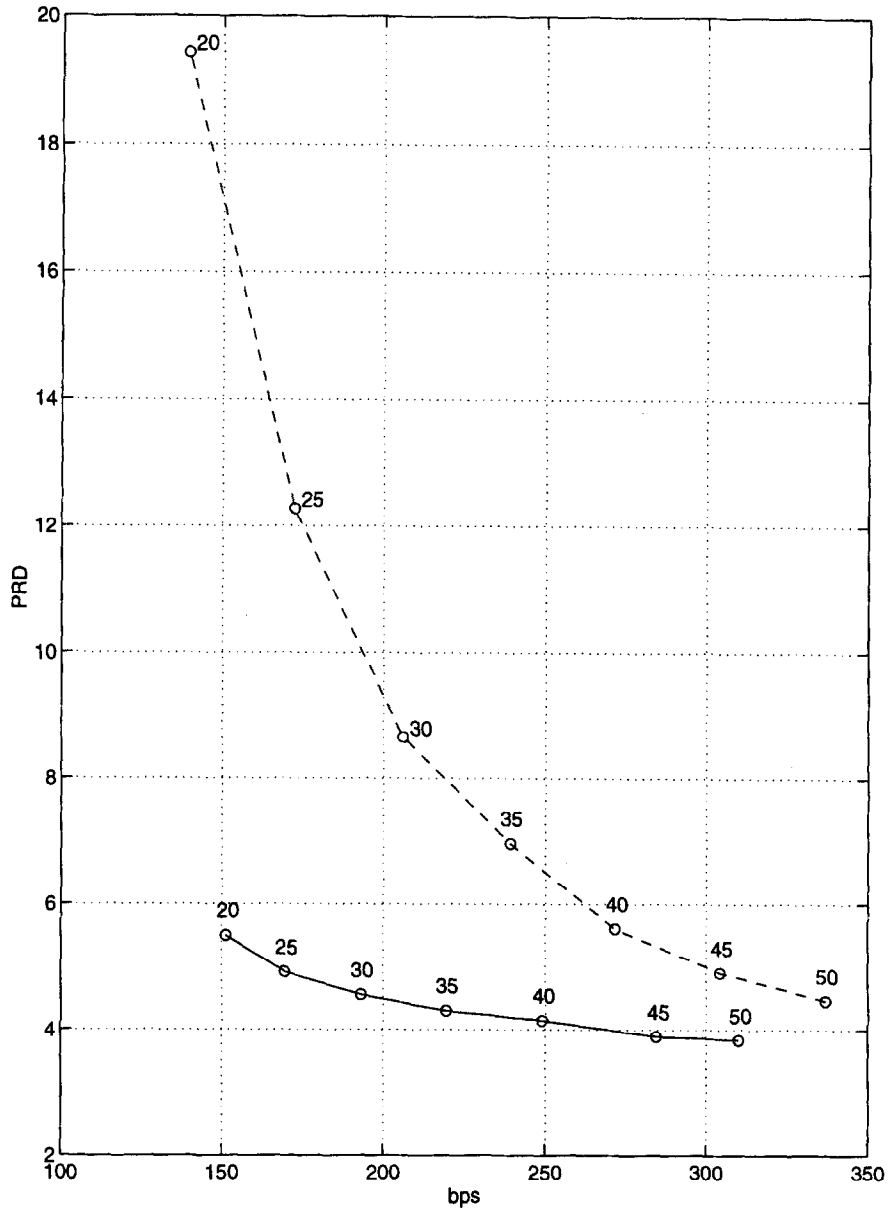

Fig. 1. Error versus bit-rate for DLT (dotted line) and B-spline basis functions (solid line).

in PRD. Therefore, visual inspection is very important. Fig. 2 depicts an example. The original signal is shown in Fig. 2(a), its approximation by 35 basis functions of DLT (PRD = 10.58, bit-rate 219.32 bps) in Fig. 2(b) and its approximation by 25 B-spline basis functions (PRD = 3.07, rate 174.82 bps) in Fig. 2(c). Very annoying ringing effect can be seen in the DLT reconstruction, while the reconstruction by spline approximation is still very good. The reconstruction errors of the proposed method and DLT are depicted in Figs. 3(a) and (b), respectively. The error of DLT is much larger than that of B-spline approximation,

Table 2
Percentage root mean square differences, bit-rates and number of bits for different number of coefficients for an ECG signal sampled at 360 Hz with 11 bits per sample resolution

n	PRD_{mean}	BR_1	BR_2	Bits per knot	Bits for the knot sequence
50	3.83	293.46	310.01	2.98	145.83
45	3.89	264.85	284.48	3.08	135.70
40	4.14	235.04	248.99	3.29	128.40
35	4.29	205.39	219.21	3.51	119.46
30	4.54	179.12	193.11	3.72	107.90
25	4.91	153.06	169.61	4.02	96.53
20	5.49	127.27	151.37	4.34	82.43

Note: PRD_{mean} is the mean value of PRD, n is the number of coefficients, BR_1 and BR_2 are the bit-rates without and with overhead, respectively.

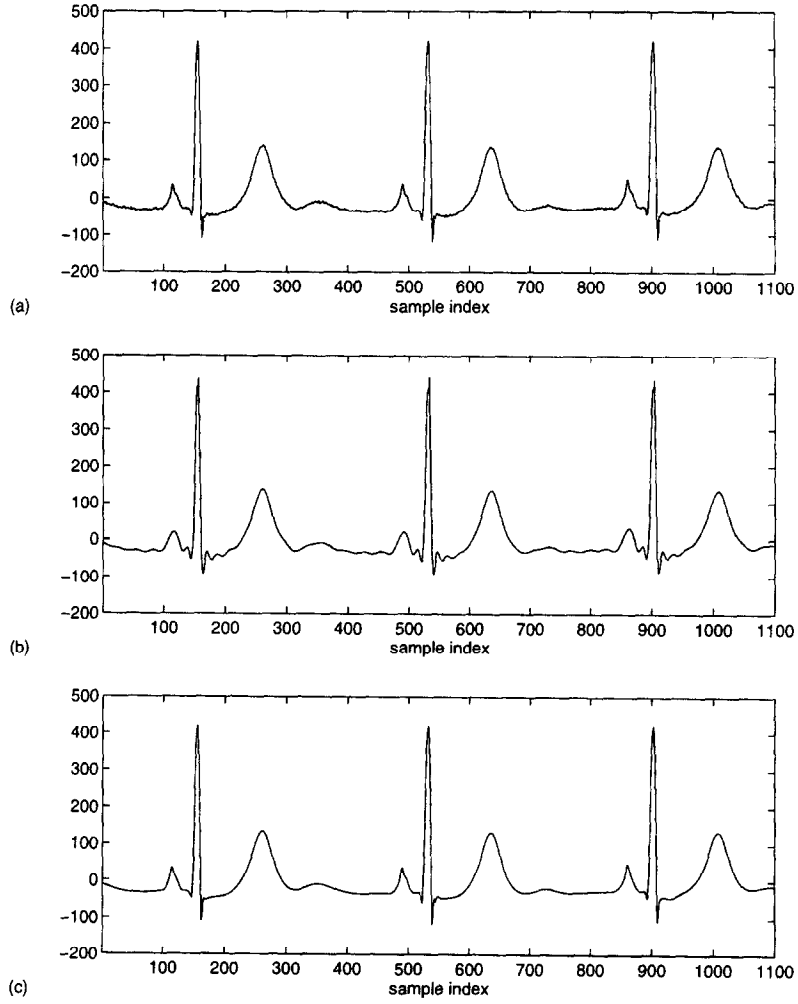

Fig. 2. (a) Original ECG (360 Hz). Its reconstruction using (b) 35 quantized coefficients of DLT (PRD = 10.58, rate 219.32 bps) and (c) 25 quantized coefficients of B-splines (PRD = 3.07, rate 174.82 bps).

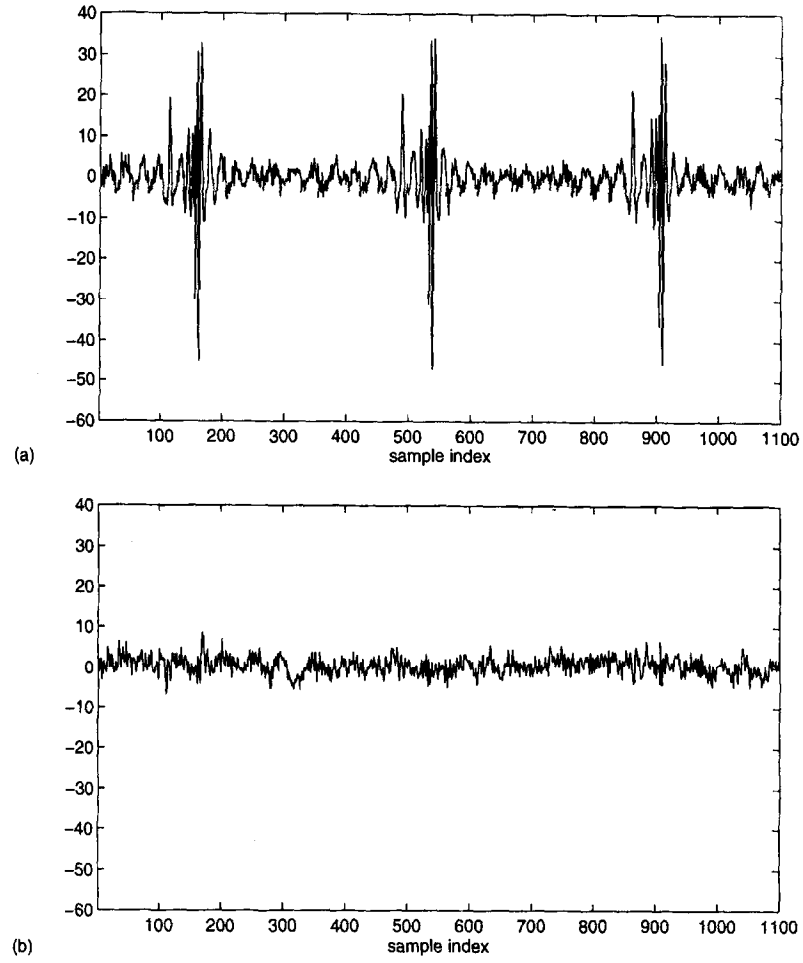

Fig. 3. Reconstruction error for the signal in Fig. 2 compressed by (a) DLT, (b) spline approximation.

especially near the QRS complexes, even though the DLT bit-rate is 25% larger.

An illustration of the knot placement and the resulting basis functions for the R–R interval is depicted in Fig. 4. Note that the local frequency of these basis functions resembles well the non-stationary behavior of the ECG signal.

Next, the same ECG signals are compressed as described in Section 4 without the restriction of constant number of basis functions. Based on the previous experiment, we choose as the number of basis functions which is adequate to represent with high quality most of the R–R complexes, $n_{\text{const}} = 25$. The PRD is now equal to 4.77 with the bit-rate 180.1 bps. An ECG record with an abnormal complex is shown in Fig. 5(a) and its spline approximation in Fig. 5(b). The abnormal complex is reconstructed with 35 basis functions.

Errors in QRS detection lead to visible artifacts in the compressed signal and large increase in bit-rate in many transform-based methods [21]. For spline compression, the quality of the reconstructed signal is not affected by the interval positioning because of the constant control over the absolute difference between the original data and its approximation. The discontinuities at the boundaries between different intervals are avoided by interpolating not approximating the first and last

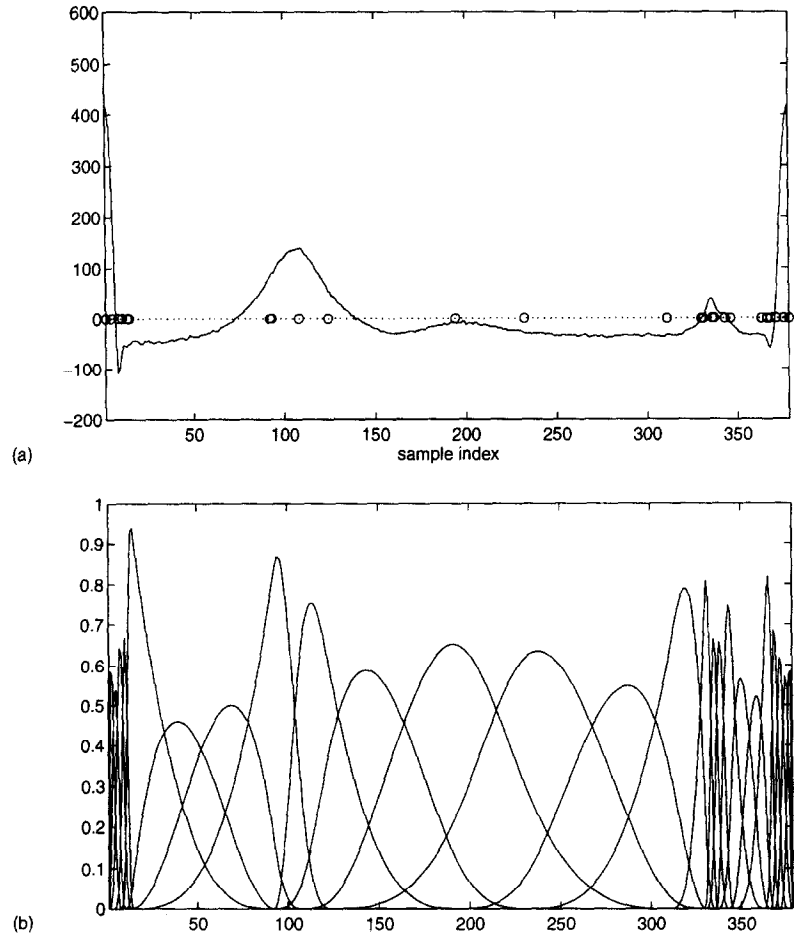

Fig. 4. (a) Location of knots (marked as circles) for the R–R interval and (b) the resulting B-spline basis functions.

points in each interval. The resulting additional bit-rate is not significant. The B-spline basis functions are able to adapt very efficiently to different types of signals; thus, there is no increase in the number of coefficients per sample. However, the false R–R complex has low correlation with intervals in the codebook and the new sequence of knots has to be sent, thereby increasing the overhead data.

6. Conclusions

A novel compression scheme based on B-spline basis functions is proposed. In addition to their ability to adapt to different signals, these basis functions can efficiently code time-varying non-stationary signals such as ECG. An iterative algorithm determines the smallest number of basis functions (shortest knot sequence, and their positions) such that the difference between the original data and its approximation remains below a given tolerance. The proposed method clearly outperforms non-adaptive DLT and DCT compression schemes. It should be noted however that the proposed adaptive algorithm is more computationally complex than these non-adaptive methods. For example, the proposed compression method takes almost twice the time required by the DLT (implementations of both methods were not optimized for speed).

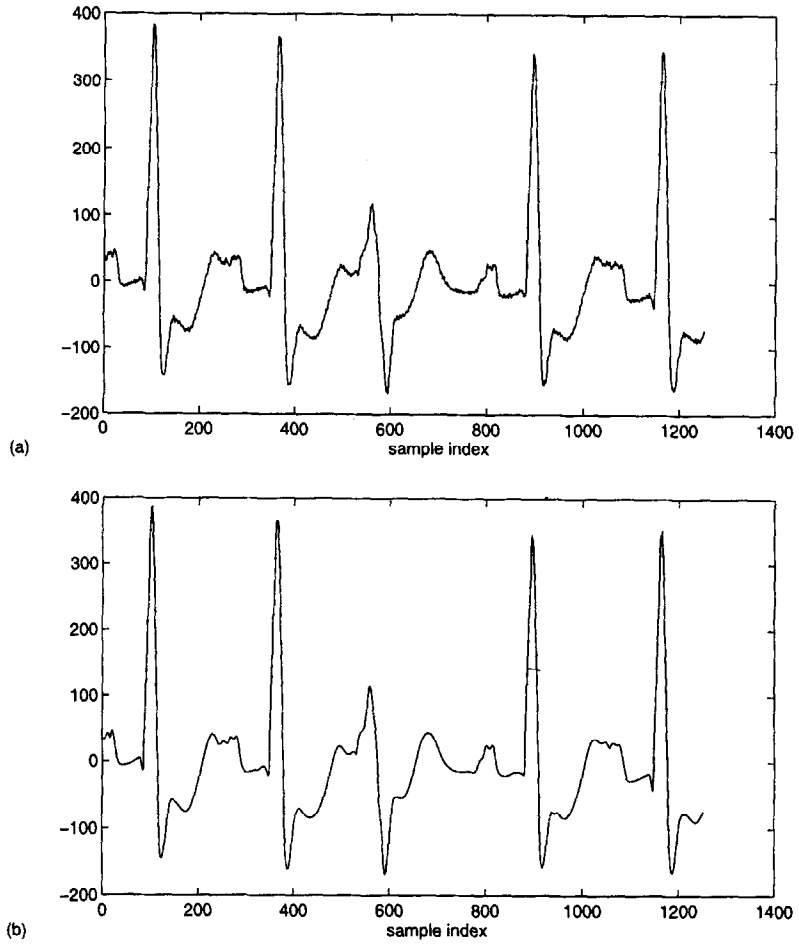

Fig. 5. (a) Original ECG signal and (b) compressed with B-spline basis functions. The abnormal complex is compressed using 35 basis functions.

References

[1] N. Ahmed, P.J. Milne, S.G. Harries, Electrocardiographic data compression via orthogonal transforms, IEEE Trans. Biomed. Eng. BME-37 (6) (1978) 484–492.

[2] W. Boehm, Inserting new knots into B-spline curves, Comput. Aided Des. 12 (4) (1980) 199–201.

[3] C. de Boor, On calculating with B-splines, J. Approx. Theory 6 (1972) 50–62.

[4] C. de Boor, J.R. Rice, Least-squares cubic spline approximation II: variable knots, Report CSD TR 21, Purdue University, West Lafayette, IN, 1968.

[5] M.G. Cox, The least-squares solution of overdetermined linear equations having band or augmented bandstructure, IMA J. Numer. Anal. 1 (1981) 3–22.

[6] P. Dierckx, Curve and Surface Fitting with Splines, Oxford University Press, New York, 1993.

[7] W.A.H. Engelse, C. Zeelenberg, A single scan algorithm for QRS-detection and feature extraction, in: IEEE Comput. Cardiology, IEEE Computer Society, Long Beach, 1979, pp. 37–42.

[8] G.H. Golub, C.F. van Loan, Matrix Computations, Johns Hopkins University Press, Baltimore, 1989.

[9] H. Imai, N. Kimura, Y. Yoshida, An efficient encoding method for electrocardiography using spline functions, System Comput. Japan 16 (3) (1985) 85–94.

[10] S.M.S. Jalaleddine, C.G. Hotchens, R.D. Strattan, W.A. Coberly, ECG data compression techniques – a unified approach, IEEE Trans. Biomed. Eng. BME-37 (4) (1990) 329–343.

[11] D.L.B Jupp, Approximation to data with splines with free knots, SIAM J. Numer. Anal. 15 (1978) 328–343.
[12] M. Karczewicz, M. Gabbouj, J. Astola, Robust B-spline image smoothing, Proc. IEEE Internat. Conf. on Image Processing, Austin, USA, November 1994, pp. 540–544.
[13] M. Karczewicz, M. Gabbouj, I. Defée, Edge detection using robust B-spline, Proc. NORSIG-94, Nordic Signal Processing Symposium, Ålesund, Norway, 2–4 June 1994, pp. 190–195.
[14] M. Karczewicz, M. Hämäläinen, M. Gabbouj, Multiple knot spline approximation for ECG data compression, Proc. IEEE Workshop on Nonlinear Signal and Image Processing, Neos-Marmaras, Halkidiki, Greece, 20–22 June 1995, pp. 487–490.
[15] W.S. Kuklinski, Fast Walsh transform data compression algorithm for ECG application, Med. Biol. Eng. Comput. 21 (1983) 465–473.
[16] G. Lachiver, J.M. Eichner, F. Besette, W. Seufert, An algorithm for ECG data compression using spline functions, Comput. Cardiol. (1986) 575–578.
[17] T. Lyche, K. Morken, A discrete approach to knot removal and degree reduction algorithms for splines, in: Algorithms for Approximation, Clarendon Press, Oxford, 1987.
[18] T. Lyche, K. Morken, A data-reduction strategy for splines with applications to the approximation of functions and data, IMA J. Numer. Anal. 8 (1988) 185–208.
[19] Massachusetts Institute of Technology, The MIT-BIH Arrhythmia Database, August 1980.
[20] W. Philips, ECG data compression with time-warped polynomials, IEEE Trans. Biomed. Eng. BME-40 (11) (1993) 1095–1101.
[21] W. Philips, G.D. Jonghe, Data compression of ECG's by high-degree polynomial approximation, IEEE Trans. Biomed. Eng. BME-39 (4) (1992) 330–337.
[22] T. Raita-aho, T. Saramäki, O. Vainio, A digital filter chip for ECG signal processing, IEEE Trans. Instrum. Measurements 43 (4) (1994) 644–649.
[23] B.R.S. Reddy, A.S.N. Murthy, ECG data compression using Fourier descriptors, IEEE Trans. Biomed. Eng. BME-33 (4) (1968) 428–434.
[24] H.I. Shahein, H.M. Abbas, ECG data compression via cubic-splines and scan-along polygonal approximation, Signal Processing 35 (3) (1994) 269–283.
[25] M. Unser, A. Aldroubi, M. Eden, B-spline signal processing: Part I – Theory, IEEE Trans. Signal Process. 41 (2) (1993) 821–833.
[26] M. Unser, A. Aldroubi, M. Eden, B-spline signal processing: Part II – Efficient design and applications, IEEE Trans. Signal Process. 41 (2) (1993) 834–848.