

Compressão de Sinais de Eletrocardiograma utilizando Recorrência de Padrões Multiescalas com Critério de Continuidade Interblocos e Segmentação Flexível

Eddie B. L. Filho, Waldir S. S. Júnior, Murilo B. de Carvalho e Eduardo A. B. da Silva

Resumo—Neste trabalho, utiliza-se o algoritmo MMP (*Multidimensional Multiscale Parser*) para a compressão de sinais de eletrocardiograma (ECG) da base de dados de arritmia MIT/BIH. O MMP é baseado no *casamento aproximado de padrões multiescalas*, codificando segmentos do sinal de entrada através de expansões e contrações de padrões previamente codificados e armazenados em um dicionário, que é construído durante o processo de compressão. Como os sinais de ECG apresentam um comportamento aproximadamente periódico, sugere-se que um esquema de compressão baseado em recorrência de padrões (principal característica do MMP) possa apresentar resultados satisfatórios, fato este que motivou o presente trabalho. Entretanto, para se obter uma compressão eficiente, várias modificações foram realizadas, tais como o critério de continuidade e a estratégia de segmentação flexível. Os resultados apresentados pelo esquema proposto foram bastante encorajadores, superando alguns dos melhores codificadores presentes na literatura.

Palavras-Chave—Casamento de Padrões Recorrentes, Decomposição Multiescalas, Quantização Vetorial, Eletrocardiograma.

Abstract—In this work, we use the MMP (*Multidimensional Multiscale Parser*) algorithm to compress electrocardiographic signals (ECG) from the MIT/BIH arrhythmia database. The MMP is based on *approximate multiscale pattern matching*, encoding segments of the input signal with expansions and contractions of patterns already encoded and stored in a dictionary, which is built during the compression process. Given that ECG signals present a quasi-periodic behavior, it is reasonable that a compression method based on recurrent patterns (the main feature of MMP) can present good results; this is the main motivation for this work. However, in order to obtain an efficient compression, several adaptations had to be performed, such as the continuity criterion and the flexible segmentation strategy. The results presented by the proposed method were very encouraging, overcoming some of the best encoders known in the literature.

Keywords—Recurrent Pattern Matching, Multiscale Decomposition, Vector Quantization, Electrocardiogram.

Eddie B. L. Filho^{*†}, Waldir S. S. Júnior[†], Murilo B. de Carvalho[‡] e Eduardo A. B. da Silva[†]. E-mails: efilho@genius.org.br, waldirjr@fucapi.br, murilo@telecom.uff.br, eduardo@lps.ufrj.br. Este trabalho foi realizado no âmbito do convênio UFAM-UFRJ/COPPE, com financiamento da SUFRAMA.

^{*}Genius Instituto de Tecnologia, Av. Açaí, 875, Bl. E, Dist. Indust., Manaus - AM, 69075-904, BRASIL. [†]PEE/COPPE/DEL/POLI, Universidade Federal do Rio de Janeiro, Cx. P. 68504, Rio de Janeiro - RJ, 21945-970, BRASIL. [‡]Fundação Centro de Análise, Pesquisa e Inovação Tecnológica, Av. Gov. Danilo de Matos Areosa, 381, Dist. Indust., Manaus - AM, 69075-351, BRASIL. [§]TET/CTC, Universidade Federal Fluminense, R. Passos da Pátria, 156, Niterói, RJ, 24210-240, BRASIL.

I. INTRODUÇÃO

Nos últimos anos, os meios de armazenamento de informações digitais têm se tornado cada vez mais baratos e apresentado um aumento expressivo de sua capacidade, o que poderia indicar uma diminuição no que diz respeito à necessidade de compressão dos dados. Entretanto, as informações tratadas também têm se tornado cada vez mais volumosas e, dependendo da aplicação, ainda há a necessidade de transmissão destes dados utilizando-se redes de telecomunicações, tais como a rede telefônica ou a rede celular. Nestes casos, dada a restrição de banda inerente aos sistemas, a compactação dos dados ainda é uma meta importante a ser atingida, empregando-se um esforço considerável para a obtenção de uma alta taxa de compressão, mantendo-se a distorção no mínimo valor possível para que a integridade dos dados não seja comprometida.

Os sinais de Eletrocardiograma (ECG), por exemplo, podem ser transmitidos utilizando-se linhas telefônicas, permitindo ao médico efetuar um diagnóstico do paciente sem que este precise se deslocar até o hospital. Esta aplicação, dada a restrição de banda de uma chamada telefônica, necessitaria de um esquema de compressão eficiente, o que permitiria até mesmo a transmissão de um exame completo com 12 canais.

O presente trabalho propõe uma nova técnica para a compressão de sinais de ECG utilizando-se o algoritmo MMP (*multidimensional multiscale parser*) [1], que é um método de compressão com perdas universal baseado no *casamento aproximado de padrões multiescalas* [1]. Neste caso, dois vetores u e v de comprimentos diferentes ($\ell(u) \neq \ell(v)$) podem ser aproximados. Para que isso seja possível, o comprimento dos vetores é modificado através de uma *transformação de escala* $T^N(\mathbf{x}) : \mathbb{R}^{\ell(\mathbf{x})} \mapsto \mathbb{R}^N$, que é implementada utilizando-se operações de mudança de taxa de amostragem [5].

Para que uma comparação com outros métodos ([3] e [4]) seja possível, serão utilizados sinais da base de dados de arritmia MIT/BIH, que contém exames de ECG de 48 indivíduos, com duas derivações de aproximadamente 30 minutos cada. Os sinais foram amostrados a 360 Hz e quantizados com 11 bits. Um desses sinais (os primeiros três segundos) está ilustrado na Figura 1. Os resultados são avaliados utilizando-se a métrica de PRD (*Percent Root mean square Difference*) e o CDR (*Compressed Data Rate*), definidos como:

$$PRD = 100 \sqrt{\frac{\sum_{n=0}^{N-1} (x(n) - \hat{x}(n))^2}{\sum_{n=0}^{N-1} (x(n) - \mu)^2}} \quad (1)$$

$$CDR = \frac{B}{T} \quad (2)$$

onde μ é a linha base do conversor analógico-digital utilizado para a aquisição dos dados $x(n)$ (no MIT/BIH $\mu = 1024$), B é o número total de bits do dado comprimido e T é a duração do sinal original, em segundos.

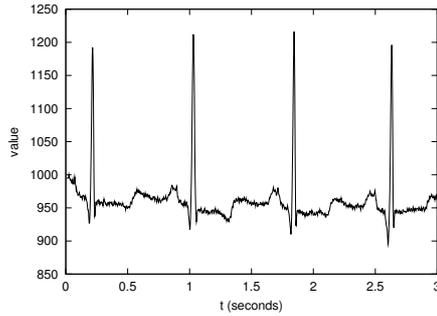


Fig. 1. Um sinal de ECG típico.

O restante deste artigo está organizado conforme descrito a seguir. Na seção II, a versão original do algoritmo MMP é apresentada. Na seção III, propõe-se a nova estratégia de segmentação flexível. Na seção IV, é apresentado o critério de continuidade. Na seção V, algumas modificações secundárias são comentadas. Resultados experimentais com sinais de ECG da base de dados MIT/BIH são apresentados na seção VI e a seção VII apresenta as conclusões do trabalho.

II. O ALGORITMO MMP

O algoritmo MMP é um esquema de compressão com perdas que utiliza a técnica de *casamento aproximado de padrões multiescalas*, uma extensão do já conhecido casamento aproximado de padrões [1], no qual vetores de dimensões diferentes podem ser aproximados.

O MMP possui um dicionário $\mathcal{D} = \{\mathbf{v}_0, \mathbf{v}_1, \dots, \mathbf{v}_{L-1}\}$ com L vetores \mathbf{v}_i de comprimento fixo, que o mesmo utiliza para codificar os segmentos de um vetor de entrada $\mathbf{X}^0 = (x(0) \ x(1) \ \dots \ x(N-1))$, cuja dimensão N é uma potência de 2. Ao tentar codificar o segmento de entrada \mathbf{X}^0 , o MMP procura no dicionário \mathcal{D} pelo melhor vetor \mathbf{v}_{i_0} para aproximar \mathbf{X}^0 . A escolha do melhor vetor é baseada na minimização do erro quadrático $\xi_0 = \|\mathbf{X}^0 - \mathbf{v}_{i_0}\|^2$ e, se o mesmo for menor ou igual a um dado limiar d^* , a codificação de \mathbf{X}^0 é realizada e o MMP retorna um *flag* '1', seguido do índice de dicionário i_0 . Caso o limiar não seja atingido ($\xi_0 > d^*$), o MMP divide o segmento de entrada em outros dois segmentos, $\mathbf{X}^1 = (x(0) \ x(1) \ \dots \ x(N/2-1))$ e $\mathbf{X}^2 = (x(N/2) \ x(N/2+1) \ \dots \ x(N-1))$, retornando um *flag* '0' e repetindo o processo de codificação para \mathbf{X}^1 . Se ocorrer um casamento, o algoritmo retorna o *flag* '1', seguido pelo seu índice de dicionário i_1 , passando então a tentar codificar o segundo segmento, \mathbf{X}^2 . Caso contrário, o MMP retorna o *flag* '0' e divide \mathbf{X}^1 em outros dois segmentos, $\mathbf{X}^3 = (x(0) \ x(1) \ \dots \ x(N/4-1))$ e $\mathbf{X}^4 =$

$(x(N/4) \ x(N/4+1) \ \dots \ x(N/2-1))$, antes de tentar codificar \mathbf{X}^2 . A divisão dos segmentos é recursivamente aplicada, até que haja um casamento ou os segmentos resultantes tenham comprimento $\ell(\mathbf{X}^j) = 1$. A Figura 2 ilustra o processo de segmentação. Neste exemplo, a saída gerada pelo MMP consistiria na sequência $0, 0, 1, i_3, 0, 1, i_9, 1, i_{10}, 1, i_2$.

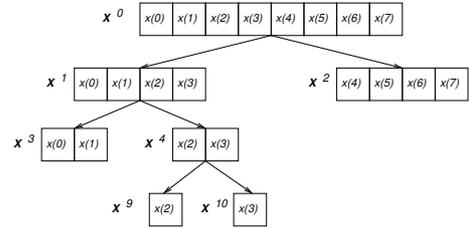


Fig. 2. Segmentação de um vetor de entrada \mathbf{X}^0 .

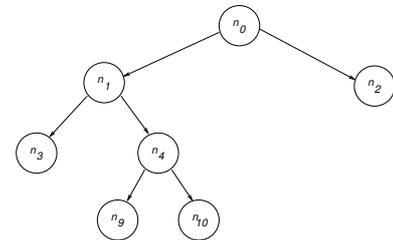


Fig. 3. Árvore de segmentação \mathcal{S} do vetor de entrada \mathbf{X}^0 da Figura 2.

A segmentação do bloco de entrada \mathbf{X}^0 é representada por uma árvore de segmentação \mathcal{S} , conforme ilustrado na Figura 3. Cada nó n_j de \mathcal{S} está associado a um segmento \mathbf{X}^j do segmento de entrada, de tamanho $(2^{-p}N)$, onde p é a profundidade do nó n_j na árvore de segmentação \mathcal{S} . Um nó n_j da árvore de segmentação pode ter ou não filhos. No caso dos mesmos existirem, suas representações são n_{2j+1} e n_{2j+2} . Um nó que não tem filhos é chamado de nó-folha e consiste numa codificação efetiva do sinal, ou seja, apenas os nós-folhas da árvore de segmentação estão associados a elementos no dicionário.

O casamento de elementos de comprimento variável com elementos de comprimento fixo é possível através de uma transformação de escala $T_N^M: \mathbb{R}^N \mapsto \mathbb{R}^M$, dada pelas fórmulas (3) e (4), que ajusta as dimensões dos vetores antes do casamento ser executado [1].

Para se transformar \mathbf{v} de comprimento N_0 em \mathbf{v}^S de comprimento $N > N_0$, $\mathbf{v}^S(n)$ ($n = 0, 1, \dots, N-1$) deve ser dado por:

$$\begin{aligned} m^0(n) &= \left\lfloor \frac{n(N_0-1)}{N-1} \right\rfloor \\ m^1(n) &= (n(N_0-1)) \bmod (N-1) \\ m^2 &= (N-N_0) \bmod (N_0-1) \\ r &= \frac{N-N_0}{N_0-1} \\ v^s(n) &= \begin{cases} v(m^0(n)), & \text{para } cd1 \\ \left\lfloor \frac{v(m^0(n)+1) - v(m^0(n))}{r+1} \right\rfloor + v^S(n-1), & \text{para } cd2 \\ \left\lfloor \frac{v(m^0(n)+1) - v(m^0(n))}{r+m^2+1} \right\rfloor + v^S(n-1), & \text{n.d.a} \end{cases} \\ cd1 &: m^1(n) = 0 \\ cd2 &: (m^1(n) > 0) \text{ and } ((m^2 = 0) \text{ or } (m^0(n) < (N_0 - 2))) \end{aligned} \quad (3)$$

Para se alterar o comprimento de N_0 para $N < N_0$, $\mathbf{v}^S(n)$ ($n = 0, 1, \dots, N-1$) deve ser dado por:

$$u(k, n) = \begin{cases} v(k + n \lfloor \frac{N_0}{N} \rfloor + 1), & \text{para cd1} \\ v(k + n \lfloor \frac{N_0}{N} \rfloor - 1), & \text{para cd2} \\ v(k + n \lfloor \frac{N_0}{N} \rfloor), & \text{n.d.a.} \end{cases}$$

$$v^S(n) = \sum_{k=-1}^{\lfloor \frac{N_0}{N} \rfloor} u(k, n)$$

$$\text{cd1} \quad : k + n \lfloor \frac{N_0}{N} \rfloor < 0$$

$$\text{cd2} \quad : k + n \lfloor \frac{N_0}{N} \rfloor > (N_0 - 1) \quad (4)$$

O dicionário do MMP é atualizado da seguinte maneira: sempre que as aproximações $\hat{\mathbf{X}}^{2j+1}$ e $\hat{\mathbf{X}}^{2j+2}$ associadas aos nós-filhos n_{2j+1} e n_{2j+2} estão disponíveis, o MMP forma um elemento aproximado $\hat{\mathbf{X}}^j$ para o bloco associado ao nó pai n_j , utilizando a concatenação das aproximações dos blocos associados aos dois nós-filhos. No exemplo da Figura 2, quando $\hat{\mathbf{X}}^9$ e $\hat{\mathbf{X}}^{10}$ estiverem disponíveis, é possível concatená-los e então obter uma nova aproximação $\hat{\mathbf{X}}^4$. Esta nova aproximação pode ser incluída no dicionário e reutilizada no futuro. Se o comprimento do vetor de entrada for N , o procedimento de divisão pode criar vetores de comprimentos $N/2, N/4, \dots, 1$. Isto implica que existem, no máximo, $1 + \log_2(N)$ diferentes comprimentos ou escalas. Portanto, para se obter uma implementação mais eficiente, basta manter $1 + \log_2(N)$ cópias do dicionário, uma em cada escala, para evitar a realização da transformação a cada nova tentativa de casamento.

A árvore de segmentação \mathcal{S} pode ser otimizada num sentido taxa-distorção, permitindo uma distribuição dos bits para a codificação que leva em consideração as necessidades globais de cada bloco de entrada. O procedimento básico de otimização, como descrito em [1], começa com uma árvore de segmentação completa (inclusive as atualizações no dicionário) e segue dos nós-folhas para o nó-raiz, podando o par de nós-filhos n_{2j+1} e n_{2j+2} sempre que o custo Lagrangeano [12] da árvore que contém os mesmos for maior que o custo da árvore de segmentação sem n_{2j+1} e n_{2j+2} . O custo Lagrangeano da árvore de segmentação \mathcal{S} é definido como $J(\mathcal{S}) = D(\mathcal{S}) + \lambda R(\mathcal{S})$, onde $D(\mathcal{S})$ é a distorção obtida utilizando-se \mathcal{S} e $R(\mathcal{S})$ é a taxa.

É importante ressaltar que sempre que um par de nós n_{2j+1} e n_{2j+2} for podado, as atualizações do dicionário resultantes da sua concatenação devem ser retiradas. Além disso, de modo a calcular corretamente os custos Lagrangeanos, é necessário manter-se um registro do número de vezes que cada vetor foi utilizado, assim como o número de ocorrências de cada *flag*.

III. A SEGMENTAÇÃO FLEXÍVEL

Conforme apresentado na seção II, a árvore de segmentação pode ser otimizada, podando-se os nós-filhos sempre que o seu custo Lagrangeano for maior que o do nó-pai. Todas as possibilidades de segmentação assim geradas, para um vetor de entrada de comprimento 8, estão ilustradas na figura 4(a),

sendo que as Figuras 4(b-c) mostram outras possibilidades de segmentação. Uma breve análise destas figuras indica que a estrutura da árvore binária restringe as opções de segmentação, pois não há como avaliar custos de nós que não tenham o mesmo nó-pai.

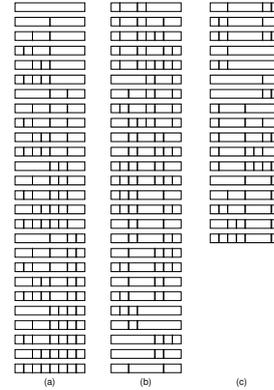


Fig. 4. Segmentação no MMP: (a) Permitido, (b-c) Não permitido.

Em [9], é apresentado um algoritmo para estender as opções de segmentação de uma árvore binária, chamado de algoritmo *prune-join* (poda-junta). Neste, um vetor de entrada é segmentado, obedecendo-se a um critério taxa-distorção [8], e cada segmento é aproximado por uma função polinomial. O primeiro passo consiste na já conhecida *poda* dos nós. O segundo, a *união*, é realizado testando-se todos os nós vizinhos que não são descendentes de um mesmo nó-pai, para verificar se a sua codificação como um único elemento reduz o custo Lagrangeano global. Então, os coeficientes dos polinômios associados aos nós restantes são quantizados e codificados. Em [9], provou-se que o algoritmo assim descrito supera a otimização baseada somente em *poda* de nós, proporcionado um desempenho taxa-distorção aproximadamente ótimo. Esses resultados teóricos não podem ser diretamente aplicados ao MMP, pois a utilização do dicionário e a abordagem baseada em recorrência de padrões multiescalas são muito diferentes da codificação escalar de polinômios utilizada no algoritmo *prune-join* original. Entretanto, espera-se um aumento no desempenho ao se estender as possibilidades de segmentação. Todas as opções de segmentação mostradas na Figura 4(b-c) podem ser obtidas de uma segmentação binária, seguida de uma união dos nós com nós-pais diferentes (nós-primos).

Para se incorporar o conceito *prune-join* ao MMP, aplica-se a segmentação original a um vetor de entrada \mathbf{X}^0 , obtendo-se uma árvore de segmentação otimizada no sentido taxa-distorção \mathcal{S} . A otimização descrita na seção II realiza a parte de *poda* do algoritmo. Depois disso, realiza-se uma análise para verificar se dois nós-primos podem ser unidos, resultando numa redução do custo Lagrangeano. A Figura 5(a) ilustra um exemplo de árvore de segmentação após a *poda*. A Figura 5(b) ilustra uma possível operação de *união*.

Define-se $U(n_j, n_i)$ como a união de dois nós-primos n_j e n_i . Em cada par destes, verifica-se se o custo de sua codificação de forma independente (isto é, $J(n_j) + J(n_i) + \lambda R_{nu}(i, j)$) é maior que o custo da codificação da sua união ($R_{nu}(i, j)$ é a taxa associada à codificação dos *flags* indicando

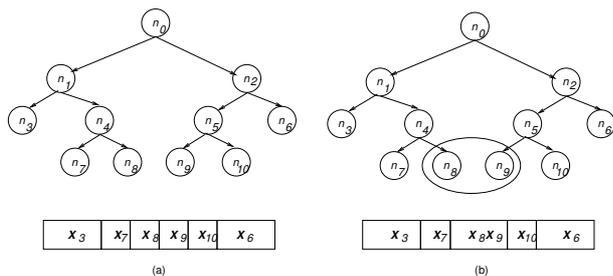


Fig. 5. O processo de união dos nós.

que os nós não devem ser unidos). O custo lagrangeano da união é dado por:

$$J(U(n_j, n_l)) = d\left(\left(\mathbf{X}^j \quad \mathbf{X}^l\right), \mathbf{v}_{i_{jl}}^s\right) + \lambda(R_u(j, l) - \log_2(\Pr(i_{jl}))) \quad (5)$$

onde $R_u(j, l)$ é a taxa gasta para se codificar o *flag* de união, $\mathbf{v}_{i_{jl}}^s$ é a melhor aproximação do dicionário para a união de \mathbf{X}^j e \mathbf{X}^l e i_{jl} representa o índice do vetor escolhido.

É interessante ressaltar que se forem utilizadas múltiplas cópias do dicionário, não haverá mais apenas $1 + \log_2(N)$ escalas diferentes, pois o procedimento de união adicionará algumas escalas extras.

IV. O CRITÉRIO DE CONTINUIDADE INTERBLOCOS

A representação $\hat{\mathbf{X}}^0$ que o MMP produz pode apresentar descontinuidades nas bordas dos segmentos $\hat{\mathbf{X}}^j$ gerados pelo procedimento de segmentação, mesmo que o vetor original \mathbf{X}^0 seja suave, fato este que está ilustrado na Figura 6.

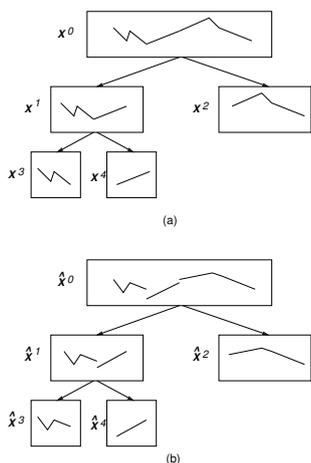


Fig. 6. Descontinuidades no MMP: (a) Sinal original; (b) Sinal reconstruído.

Isto ocorre porque a distorção calculada, usada para avaliar o custo do nó n_j , é independente da representação escolhida para os segmentos vizinhos ao atual. Em [10], foi proposto um método eficiente para se controlar a suavidade de representações geradas por esquemas de quantização vetorial, chamado de *side-match vector quantization* (SM-VQ). O SM-VQ pode ser incorporado ao MMP de modo a melhorar seu desempenho para sinais de entrada suaves [11]. A idéia é escolher um subconjunto do dicionário \mathcal{D} , chamado de

dicionário de estado \mathcal{D}_S , composto pelos N_S melhores vetores $\mathbf{v}_k \in \mathcal{D}$ dados pelo critério de continuidade adotado. Para se compreender melhor o critério de continuidade no MMP, são necessárias algumas definições.

A posição da primeira amostra de \mathbf{X}^j dentro de \mathbf{X}^0 é dada por:

$$Fp(j) = N \left((j+1) 2^{-\lfloor \log_2(j+1) \rfloor} - 1 \right) \quad (6)$$

onde N é o comprimento de \mathbf{X}^0 .

O comprimento N^j de \mathbf{X}^j , pode ser calculado como:

$$N^j = N 2^{-\lfloor \log_2(j+1) \rfloor} \quad (7)$$

O vizinho esquerdo de \mathbf{X}^j é dado por:

$$\mathbf{L}^j = \left(\hat{x}(Fp(j) - N^j) \quad \dots \quad \hat{x}(Fp(j) - 1) \right), Fp(j) \geq 1 \quad (8)$$

De acordo com a equação (8), o vizinho esquerdo \mathbf{L}^j é um vetor de mesmo comprimento que \mathbf{X}^j , cujos componentes são as amostras reconstruídas à esquerda de \mathbf{X}^j .

Para se medir a suavidade da aproximação, definem-se três parâmetros:

1) A *descontinuidade de ordem zero*:

$$D^0(\mathbf{v}_k^s, j) = \left| L^j(N^j - 1) - v_k^s(0) \right| \quad (9)$$

2) A *descontinuidade de primeira ordem*:

$$D^1(\mathbf{v}_k^s, j) = \left| L^j(N^j - 2) - L^j(N^j - 1) - v_k^s(0) + v_k^s(1) \right| \quad (10)$$

3) A *descontinuidade de segunda ordem*:

$$D^2(\mathbf{v}_k^s, j) = \left| L^j(N^j - 3) - 2L^j(N^j - 2) + L^j(N^j - 1) - v_k^s(0) + 2v_k^s(1) - v_k^s(2) \right| \quad (11)$$

Estas definições são baseadas nas aproximações para as suas respectivas derivadas.

Define-se, então, a métrica de *rugosidade* como:

$$R(\mathbf{v}_k^s, j) = \alpha D^0(\mathbf{v}_k^s, j) + \beta D^1(\mathbf{v}_k^s, j) + \gamma D^2(\mathbf{v}_k^s, j) \quad (12)$$

onde \mathbf{v}_k^s é um vetor escalonado do dicionário.

Para que fosse possível a identificação de transições através dos blocos, possibilitando assim a continuidade do sinal e uma escolha confiável dos elementos do dicionário de estado, a métrica de *rugosidade* específica utilizada foi:

$$R(\mathbf{v}_k^s, j) = \left| \left| L^j(N^j - 3) - L^j(N^j - 1) + v_k^s(0) - v_k^s(2) \right| - \left[\frac{4}{3} \left| L^j(N^j - 2) - v_k^s(1) \right| \right] \right| \quad (13)$$

Assim, o MMP escalona todos os vetores de \mathcal{D} para um dado comprimento e constrói um dicionário de estado \mathcal{D}_S contendo os N_S vetores menos “rugosos”, de acordo com a equação (13). A Figura 7 ilustra como calcular $R(\mathbf{v}_0, 2)$.

O tamanho N_S do dicionário de estado é calculado antes de cada tentativa de codificação. Baseando-se na asserção de que sinais mais simples exigem um número menor de vetores

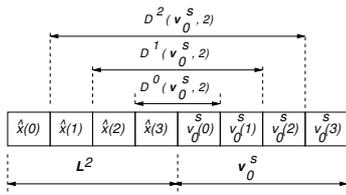
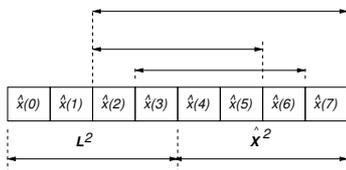


Fig. 7. Cálculo da rugosidade


 Fig. 8. As três novas atualizações ($j = 2$).

para serem bem representados [11], utiliza-se a métrica de *atividade*, como definida na equação (14), para estimar o número de vetores N_S necessários em \mathcal{D}_S para se codificar o segmento de entrada. A atividade foi escolhida por levar em consideração as bordas ou transições presentes no vetor, sendo uma medida razoável da complexidade do sinal.

$$A(\mathbf{L}^j) = \sum_{n=1}^{N^j-1} |L^j(n-1) - L^j(n)| \quad (14)$$

O tamanho final do dicionário de estado é proporcional à atividade do vizinho esquerdo, avaliada pela equação (14). Além disso, o ato de se preencher o dicionário de estado com os N_S vetores menos “rugosos” pode ser interpretado como um simples modelo estatístico para a fonte.

V. MODIFICAÇÕES COMPLEMENTARES

Como pode ser visto na Figura 1, um sinal de ECG típico é aproximadamente periódico. O algoritmo MMP, por sua vez, adapta o seu dicionário ao sinal de entrada e tem a capacidade de aprender o padrão em um período. Entretanto, haja vista que o comprimento dos segmentos podem não ser múltiplos do período do sinal, também é necessário aprender várias versões deslocadas de um período para uma representação eficiente do sinal. De modo a melhorar o desempenho para sinais aproximadamente periódicos, pode-se utilizar um *dicionário de deslocamentos* \mathcal{D}^D , o qual inclui versões deslocadas de aproximações de segmentos já codificados, aumentando a adaptabilidade do algoritmo. Tal dicionário pode ser implementado mantendo-se as M últimas amostras do sinal reconstruído em um vetor \mathbf{V}_D^j . Além disso, é necessário mais um *flag* para identificar a origem do elemento (\mathcal{D}^D ou \mathcal{D}).

O MMP utiliza o seu dicionário como um *codebook* de um quantizador vetorial. Portanto, seu desempenho deve melhorar com o aumento do mesmo. Com o objetivo de aumentar a taxa de crescimento do dicionário, adicionam-se quatro novos vetores em cada atualização, ao invés de apenas um. Quando o segmento $\hat{\mathbf{X}}^j$ é determinado, incluem-se os vetores ilustrados na Figura 8.

Antes de se codificar \mathbf{X}_m^0 , o dicionário \mathcal{D} é inicializado com $\mathcal{D}_0 = \{x_{min}, x_{min} + 4, \dots, x_{max}\}$, onde x_{min} e x_{max}

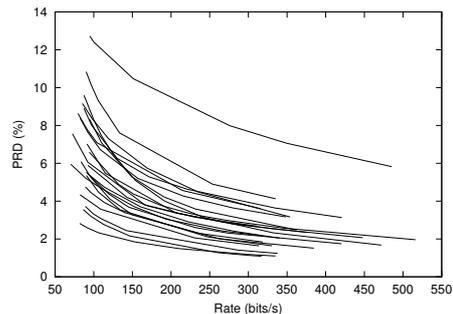
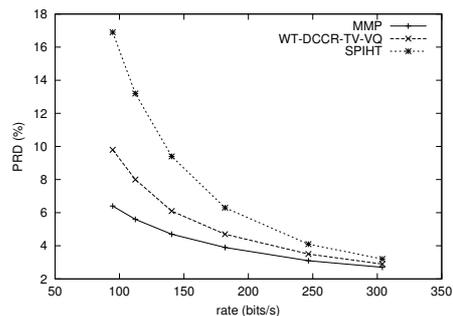

 Fig. 9. Desempenho do MMP com a base de dados MIT/BIH ($PRD \times CDR$)


Fig. 10. Desempenhos do MMP, SPIHT [3] e WT-DCCR-TV-VQ [4]

são respectivamente os valores mínimo e máximo das amostras de $x(n)$. Isto é feito para que os primeiros segmentos possam ser codificados e o dicionário \mathcal{D} comece a aprender os padrões do sinal.

VI. RESULTADOS DE SIMULAÇÕES

O MMP foi implementado em software e utilizado na compressão de sinais de ECG da base de dados de arrhythmia MIT/BIH. Com o objetivo de se realizar comparações com os algoritmos [3] e [4] (ambos baseados em DWT), aplicou-se o MMP aos dois canais dos seguintes registros: 100, 101, 102, 103, 107, 109, 111, 115, 117, 118 and 119.

A Figura 9 mostra os resultados das simulações (PRD versus *taxa de bits*) para os 11 registros escolhidos da base de dados MIT/BIH (comprimento total).

A Figura 10 mostra uma comparação com os resultados médios informados em [3] e [4]. Os resultados foram obtidos tirando-se a média dos valores de PRD obtidos para os dois canais dos 11 registros em cada taxa. Esta figura deixa claro que o MMP supera os outros dois métodos para todas as taxas apresentadas. A tabela I contém a mesma informação da Figura 10, só que em forma tabular.

A Figura 11 contém 1 segundo do canal 0 do registro 100, em diferentes valores de PRD e taxa de bits.

VII. CONSIDERAÇÕES FINAIS

Um método de compressão com perdas universal recentemente desenvolvido, chamado de MMP, foi aplicado na compressão de sinais de ECG. Este método é baseado no casamento aproximado de padrões multiescalas, uma extensão do casamento de padrões comum. O MMP utiliza um dicionário de padrões, que é construído de forma adaptativa

TABELA I
COMPARAÇÃO DE DESEMPENHO ENTRE MMP, SPIHT E WT-DCCR-TV-VQ ($PRD \pm \sigma$)

$CDR(bits/s)$	94.7	112.3	140.5	181.9	246.6	304.0
<i>MMP</i>	6.4 ± 2.45	5.6 ± 2.22	4.7 ± 1.99	3.9 ± 1.79	3.1 ± 1.57	2.7 ± 1.41
<i>WT - DCCR - TV - VQ</i>	9.8 ± 3.60	8.0 ± 2.97	6.1 ± 2.33	4.7 ± 1.86	3.5 ± 1.44	2.9 ± 1.23
<i>SPIHT</i>	16.9 ± 3.68	13.2 ± 2.95	9.4 ± 2.32	6.3 ± 1.59	4.1 ± 1.09	3.2 ± 0.94

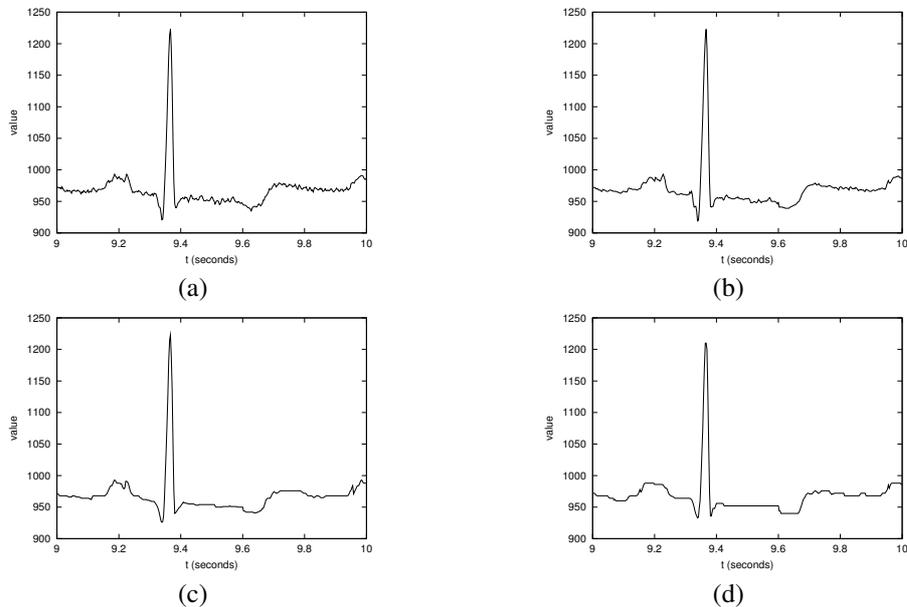


Fig. 11. Registro 100: (a) Original; (b) $PRD = 2.67$ e $CDR = 305.41$; (c) $PRD = 3.84$ e $CDR = 164.97$; (d) $PRD = 6.04$ e $CDR = 89.57$

enquanto a informação é codificada, e um procedimento de segmentação simples, que pode ser facilmente estendido para operar em N dimensões. Alguns conceitos disponíveis na literatura foram incorporados ao algoritmo básico, obtendo-se melhoras significativas no seu desempenho, fato devido principalmente à segmentação flexível e ao critério de continuidade interblocos. Um dicionário de deslocamentos também foi agregado, objetivando-se explorar a característica aproximadamente periódica dos sinais de ECG. Apesar de originalmente utilizado na compressão de imagens, o MMP apresentou ótimos resultados com relação aos sinais de ECG, superando alguns dos melhores codificadores conhecidos na literatura. Portanto, conclui-se que o MMP é um método eficiente para a compressão de sinais de ECG e abre novos caminhos para o desenvolvimento de codificadores adequados a este tipo de sinal.

AGRADECIMENTOS

Este trabalho é resultado do programa de Mestrado interinstitucional UFAM-UFRJ/COPPE, financiado pela Superintendência da Zona Franca de Manaus-SUFRAMA. Os autores agradecem à coordenadoria do programa de mestrado e à SUFRAMA pelo apoio recebido para a sua realização.

REFERÊNCIAS

- [1] M. B. de Carvalho, E. A. B. da Silva and W. A. Finamore, "Multidimensional signal compression using multiscale recurrent patterns", *Signal Processing: Image and Video Coding beyond Standards*, No. 82, pp. 1559-1580, Elsevier Science B. V., 2002.
- [2] S. Jaleldine, C. Hutchens, R. Strattan and W. Coberly, "ECG data compression techniques-A unified approach," *IEEE transactions on Biomedical Engineering*, Vol. 37, No. 4, pp. 329-343, April 1990.
- [3] 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, July 2000.
- [4] 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, July 2002.
- [5] P. P. Vaidyanathan, *Multirate Systems and Filter Banks*, Englewood Cliffs, New Jersey: Prentice-Hall Inc., 1993.
- [6] J. Ziv and A. Lempel, "Compression of individual sequences via variable-rate coding," *IEEE Transactions on Information Theory*, Vol. it-24, No. 5, pp. 530-536, September 1978.
- [7] R. A. Blahut, *Principles and Practice of Information theory*, Cambridge, Massachusetts: Addison-Wesley publishing Company, 1988.
- [8] G. J. Sullivan and R. L. Baker, "Efficient quadtree coding of images and video," *IEEE Transactions on Image Processing*, Vol. 3, No. 3, pp. 327-331, May 1994.
- [9] R. Shukla, P. L. Dragotti, M. N. Do and M. Vetterli, "Rate-Distortion Optimized Tree-Structured Compression Algorithms for Piecewise Polynomial Images," *IEEE Transactions on Image Processing*, Vol. 14, No. 3, pp. 343-359, March 2005.
- [10] T. Kim, "Side Match and Overlap Match Vector Quantizers for Images," *IEEE Transactions on Image Processing*, Vol. 1, No. 2, pp. 170-185, February 1992.
- [11] E. B. L. Filho, M. B. Carvalho and E. A. B. da Silva, "Multidimensional signal compression using multi-scale recurrent patterns with smooth side-match criterion," *IEEE International Conference on Image Processing*, Singapore, October, 2004.
- [12] M. M. Denn, *Optimization by Variational Methods*, New York: McGraw-Hill Book Company, 1969.