

UM MÉTODO PARA REDUÇÃO DA COMPLEXIDADE COMPUTACIONAL DA ETAPA DE CODIFICAÇÃO DA QUANTIZAÇÃO VETORIAL

F. Madeiro Waslon T. A. Lopes Marcelo S. Alencar Benedito G. Aguiar Neto

Laboratório de Comunicações, DEE, COPELE – Universidade Federal da Paraíba
58.109-970 - Campina Grande, PB, Brasil
{madeiro,waslon,malencar,bganeto}@dee.ufpb.br

RESUMO

Em um artigo recente [1], mostrou-se que um algoritmo baseado em análise de componentes principais, denominado PCA (*principal component analysis*), apresenta-se como uma técnica apropriada para projeto de dicionários aplicados à codificação de voz baseada em QV (quantização vetorial). No presente trabalho, a simetria existente nos dicionários PCA é explorada em um método que reduz significativamente a complexidade computacional da etapa de codificação da QV.

1. INTRODUÇÃO

Um quantizador vetorial [2, 3] K -dimensional de N níveis pode ser definido como um mapeamento Q do espaço euclidiano K -dimensional R^K em um subconjunto finito W contendo N vetores em R^K , isto é,

$$Q : R^K \rightarrow W, \quad (1)$$

em que $W = \{w_i; i = 1, 2, \dots, N\}$ é denominado dicionário e os vetores w_i são denominados vetores-códigos ou vetores de reconstrução.

Na etapa de codificação da quantização vetorial (QV), o codificador gera a palavra-binária $b_I \in \{0, 1\}^m$ para o caso de o vetor-código w_I ser o mais semelhante ao vetor de entrada (vetor a ser codificado) x dentre todos os vetores-códigos do dicionário. O codificador utiliza, para tanto, a regra do vizinho mais próximo, em que a palavra-binária b_I é transmitida se $d(x, w_I) < d(x, w_i), \forall i \neq I$, em que $d(\cdot)$ denota uma medida de distorção (distância). No outro lado do sistema de comunicação baseado em QV, o decodificador, que dispõe de uma réplica do dicionário W , ao receber a palavra b_I de m bits, simplesmente procura pelo I -ésimo vetor-código e produz o vetor w_I como a reprodução (versão quantizada) de x .

A distorção introduzida na representação dos vetores da fonte pelos correspondentes vetores-códigos depende da qualidade do dicionário W . A taxa de codificação de um quantizador vetorial, que mede o número de bits por componente de vetor, é dada por $R = \frac{m}{K} = \frac{1}{K} \log_2 N$. Em

codificação de forma de onda de voz, R é expressa em bit/amostra. Recentemente [1, 4], diversas avaliações mostraram que um algoritmo baseado em análise de componentes principais, denominado PCA (*principal component analysis*), apresenta-se como uma técnica adequada para projeto de dicionários aplicados à codificação de forma de onda de voz, tendo sido destacados aspectos que apontam uma superioridade do algoritmo PCA sobre outras técnicas de projeto, como o algoritmo LBG (Linde-Buzo-Gray) [5] e algoritmos de redes neurais [6–8]. Dentre esses aspectos, destacam-se: simplicidade e rapidez, geração de dicionários estruturados, inexistência de parâmetros a serem ajustados e independência de dicionário inicial. Testes subjetivos informais relatados em [1, 4] mostraram que, em se tratando de QV com dimensão $K = 2$, os sinais de voz reconstruídos com o uso de dicionários PCA apresentam qualidade superior à obtida com o uso de dicionários LBG para diversas taxas de codificação avaliadas.

No presente trabalho é apresentado um novo método para redução da complexidade computacional envolvida na etapa de codificação da QV. O método proposto utiliza a organização estruturada (em particular, a simetria) dos dicionários PCA para reduzir o número de multiplicações, adições, subtrações e comparações realizadas na etapa de codificação por distorção mínima (etapa de determinação do vizinho mais próximo).

Este artigo encontra-se organizado de acordo com as seções a seguir. Na Seção 2 é apresentada uma abordagem sucinta da questão da complexidade computacional da QV. A simetria dos dicionários PCA é discutida na Seção 3. A Seção 4 descreve o algoritmo proposto para redução do número de operações realizadas na etapa de codificação da QV. Resultados e conclusão são apresentados, respectivamente, nas Seções 5 e 6.

2. COMPLEXIDADE COMPUTACIONAL DA QV

Em quantização vetorial, a complexidade computacional existente na etapa de codificação dos vetores a serem quantizados constitui um problema crítico. A representação de um vetor de entrada por um dos vetores-códigos, ou seja,

a determinação do vizinho mais próximo, representa uma das principais limitações da QV: a complexidade computacional na etapa de codificação da QV cresce exponencialmente com a dimensão para uma dada taxa, conforme será demonstrado a seguir.

Para codificar um dado vetor de entrada, é necessário encontrar sua distância para cada um dos N vetores-códigos do dicionário e depois comparar as distâncias de modo a encontrar o vetor-código mais semelhante, ou seja, o vizinho mais próximo. No algoritmo convencional de busca total (BT), a codificação de cada vetor de entrada requer, portanto, N cálculos de distância (distorção) e $N - 1$ comparações. No caso da medida de distorção de erro quadrático, isto é,

$$d(\mathbf{x}, \mathbf{w}_i) = \sum_{j=1}^K (x_j - w_{ij})^2, \quad (2)$$

em que w_{ij} é a j -ésima componente do vetor-código \mathbf{w}_i e x_j é a j -ésima componente do vetor de entrada \mathbf{x} , cada cálculo de distância requer K multiplicações, K subtrações e $K - 1$ adições. Portanto, para codificar cada vetor de entrada, KN multiplicações, KN subtrações, $(K - 1)N$ adições e $N - 1$ comparações devem ser computadas. A complexidade de um quantizador vetorial pode ser alternativamente expressa em termos de N multiplicações, N subtrações e $(1 - \frac{1}{K})N$ adições e $(N - 1)/K$ comparações por amostra.

Em resumo, a complexidade computacional de um quantizador vetorial de dimensão K e taxa R requer um número de operações por amostra da ordem de $N = 2^{KR}$ para cada vetor de entrada se uma busca completa (busca total, busca exaustiva) é realizada ao longo do dicionário.

3. DICIONÁRIOS PCA

Uma das características dos dicionários PCA é o fato de serem estruturados, ou seja, terem uma disposição bastante organizada no espaço de padrões [1], conforme mostra a Figura 1. Esta característica pode ser utilizada para minimizar os requisitos de capacidade de memória exigida para armazenar o dicionário bem como para reduzir significativamente o tempo de processamento envolvido na etapa de codificação da QV (no processo de determinação do vizinho mais próximo).

Observa-se que um dicionário PCA apresenta as seguintes características:

1. Os N vetores-códigos estão ordenados de acordo com a Figura 2, em que N_{POS} , N_{ZERO} e N_{NEG} denotam os vetores-códigos com médias¹ positiva, zero e negativa, respectivamente. Além disso, N_{ZERO} representa uma pequena parcela do total de $N = N_{\text{POS}} + N_{\text{ZERO}} + N_{\text{NEG}}$ vetores-códigos do dicionário. Em muitos casos, $N_{\text{ZERO}} = 0$;

¹Ao longo deste trabalho, a média de um vetor deve ser entendida como a média de suas componentes.

2. Os vetores-códigos encontram-se dispostos em ordem decrescente de valores médios das componentes;

3. Conforme mostra a Tabela 1, existe uma simetria bastante marcante: associado a um vetor-código \mathbf{w}_i , $1 \leq i \leq N/2$, existe um vetor-código \mathbf{w}_{N+1-i} tal que $\mathbf{w}_{N+1-i} = -\mathbf{w}_i$.

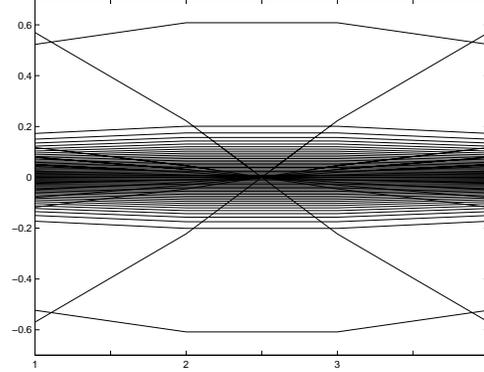


Figura 1: Dicionário PCA com $K = 4$ e $N = 64$. Cada curva do conjunto de 64 curvas é obtida conectando-se os pontos correspondentes às componentes dos vetores-códigos.

Os primeiros $N/2$ vetores-códigos do dicionário, portanto, têm componentes cujo valor médio é positivo ou zero. Esses vetores serão denotados a partir de agora por $\mathbf{w}_{i,\text{PZ}}$, com $1 \leq i \leq N/2$, em que o subscrito PZ é uma alusão ao valor médio positivo ou zero. Os últimos $N/2$ vetores-códigos do dicionário, por outro lado, têm componentes cujo valor médio é negativo ou zero. Esses vetores serão denotados a partir de agora por $\mathbf{w}_{i,\text{NZ}}$, com $N/2 + 1 \leq i \leq N$, em que o subscrito NZ é uma alusão ao valor médio negativo ou zero. Além disso, em decorrência da simetria dos dicionários PCA, associado a cada vetor-código $\mathbf{w}_{i,\text{PZ}}$ existe um correspondente vetor-código $\mathbf{w}_{N+1-i,\text{NZ}}$ tal que $\mathbf{w}_{i,\text{PZ}} = -\mathbf{w}_{N+1-i,\text{NZ}}$, $1 \leq i \leq N/2$.

Em virtude dessa simetria, é necessário armazenar apenas metade dos vetores-códigos. De fato, armazenando-se apenas os vetores-códigos $\mathbf{w}_{i,\text{PZ}}$, os vetores-códigos $\mathbf{w}_{i,\text{NZ}}$ são facilmente determinados. Com isso, reduz-se pela metade os requisitos de memória de armazenamento para o dicionário. A simetria dos dicionários PCA também contribui para a concepção de uma estratégia para redução da complexidade computacional da etapa de codificação (determinação do vizinho mais próximo) da QV, conforme mostra a seção a seguir. No método descrito no presente trabalho, apenas a metade do dicionário correspondente aos vetores $\mathbf{w}_{i,\text{PZ}}$ é efetivamente armazenada na memória de referência do codificador. O decodificador, por outro lado, dispõe dos N vetores-códigos do dicionário.

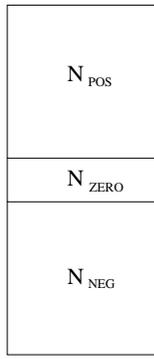


Figura 2: Organização de um dicionário PCA com $N = N_{\text{POS}} + N_{\text{ZERO}} + N_{\text{NEG}}$ vetores-códigos.

4. ALGORITMO PARA REDUÇÃO DA COMPLEXIDADE COMPUTACIONAL DA QV

Com o método de codificação introduzido no presente trabalho, a codificação de um determinado vetor de entrada (vetor da fonte a ser quantizada) x é realizada da seguinte forma: dado x , se $\text{média}(x) \geq 0$ então² o codificador realiza uma busca pelo vizinho mais próximo de x apenas nos vetores-códigos $w_{i,\text{PZ}}$, isto é, nos vetores-códigos efetivamente armazenados na memória de referência do codificador. Neste caso, é transmitida para o decodificador uma palavra-binária iniciada por 0, indicando para o decodificador que o vetor-código a ser produzido como representante de x trata-se de um vetor do tipo $w_{i,\text{PZ}}$, seguida de $\log_2(N/2)$ bits necessários para representar o índice i do vetor $w_{i,\text{PZ}}$ selecionado da memória de referência. Por outro lado, se $\text{média}(x) < 0$, a busca pelo vizinho mais próximo deveria contemplar uma procura dentre os vetores-códigos $w_{i,\text{NZ}}$. Como estes vetores não se encontram armazenados na memória de referência do codificador, a codificação consiste em comparar o vetor $-x$ (vetor simétrico de x) com os vetores-códigos $w_{i,\text{PZ}}$. Uma vez determinado o melhor representante para $-x$, é transmitida uma palavra-binária iniciada por 1, indicando que o vetor-código a ser produzido pelo decodificador como representante de x trata-se de um vetor do tipo $w_{i,\text{NZ}}$, seguida de uma seqüência de $\log_2(N/2)$ bits: cada bit dessa seqüência é o complemento do bit correspondente na seqüência de $\log_2(N/2)$ bits necessários para representar o índice i do vetor $w_{i,\text{PZ}}$ selecionado como o mais próximo de $-x$ segundo um critério de distorção mínima. É importante observar que, devido à simetria/organização do dicionário PCA, o fato de x ter como vizinho mais próximo um determinado vetor-código $w_{N+1-i,\text{NZ}}$ implica dizer que o simétrico $-x$ tem como vizinho mais próximo $w_{i,\text{PZ}} = -w_{N+1-i,\text{NZ}}$.

²Para um vetor $x = [x_1 x_2 \dots x_K]^T$, em que T denota a operação de transposição, $\text{média}(x) = \frac{1}{K} \sum_{j=1}^K x_j$.

Tabela 1: Dicionário PCA com 16 vetores-códigos bidimensionais $w_i = [w_{i1} \ w_{i2}]^T$, com $1 \leq i \leq 16$. A palavra-binária do i -ésimo vetor-código, w_i , é denotada por b_i , enquanto que w_{ij} representa a j -ésima componente do vetor w_i .

i	w_{i1}	w_{i2}	b_i
1	0,5784	0,5784	0000
2	0,1365	0,1365	0001
3	0,1020	0,1020	0010
4	0,0765	0,0765	0011
5	0,0550	0,0550	0100
6	0,0357	0,0357	0101
7	0,0176	0,0176	0110
8	0,0010	0,0010	0111
9	-0,0010	-0,0010	1000
10	-0,0176	-0,0176	1001
11	-0,0357	-0,0357	1010
12	-0,0550	-0,0550	1011
13	-0,0765	-0,0765	1100
14	-0,1020	-0,1020	1101
15	-0,1365	-0,1365	1110
16	-0,5784	-0,5784	1111

Para ilustrar o algoritmo de codificação proposto neste trabalho, considere o dicionário PCA da Tabela 1, que corresponde à tabela disponível para o decodificador. Já o codificador dispõe da Tabela 2, correspondente aos primeiros $N/2$ vetores-códigos, do tipo $w_{i,\text{PZ}}$, da Tabela 1. Suponha que o sistema de comunicação receba como entrada o vetor $x = [0,0757 \ 0,0792]^T$. Após a avaliação da média³ das componentes desse vetor, o codificador decide que o vetor-código a ser selecionado como o representante (como a versão quantizada) de x trata-se de um vetor do tipo $w_{i,\text{PZ}}$. Determina-se, assim, que o primeiro bit da palavra-binária transmitida para o decodificador é 0. A busca do vizinho mais próximo é então realizada no dicionário efetivamente armazenado (Tabela 2) na memória de referência do codificador. Seguindo o critério de distorção mínima o vetor $[0,0765 \ 0,0765]^T$, com representação binária 011, é selecionado. O codificador, portanto, transmite para o decodificador a palavra-binária 0011, em que o primeiro bit informa que a versão quantizada de x é do tipo $w_{i,\text{PZ}}$ (isto é, um dos primeiro $N/2$ vetores do dicionário de N vetores do decodificador) e os três últimos bits são uma representação binária do índice do vetor selecionado (da tabela do codificador) como vizinho mais próximo de x . No outro lado do

³Tendo em vista que soma positiva de todas as componentes de um vetor implica média positiva das componentes, em lugar de determinar a média das componentes do vetor o método de codificação proposto calcula apenas a soma das componentes. Economiza-se, assim, uma operação de divisão para cada vetor de entrada.

sistema de comunicação baseado em QV, o decodificador, que dispõe do dicionário da Tabela 1, ao receber a palavra-binária 0011, produz como saída o vetor $[0, 0765 \ 0, 0765]^T$.

Suponha, agora, que o sistema de comunicação tenha como entrada o vetor $\mathbf{x} = [-0, 4795 \ -0, 4957]^T$. Após avaliar a média desse vetor, o codificador decide que a versão quantizada de \mathbf{x} trata-se de um vetor do tipo $w_{i,NZ}$. Determina-se, assim, que o primeiro bit da palavra-binária transmitida para o decodificador é 1. É realizada uma busca pelo vizinho mais próximo de $-\mathbf{x}$ dentre os $N/2$ vetores-códigos disponíveis (Tabela 2) para o codificador: é selecionado o vetor-código $[0, 5784 \ 0, 5784]^T$ por ser o mais próximo (semelhante) a $[0, 4795 \ 0, 4957]^T = -\mathbf{x}$. O codificador, portanto, transmite para o decodificador a palavra-binária 1111, em que o primeiro bit informa que o vetor-código selecionado como a versão quantizada do vetor de entrada (vetor da fonte) \mathbf{x} é do tipo $w_{i,NZ}$ e os três últimos bits correspondem ao complemento da palavra-binária 000 que identifica o vetor $[0, 5784 \ 0, 5784]^T$ no dicionário efetivamente disponível ao codificador. No outro lado do sistema de comunicação, o decodificador, ao receber a palavra-binária 1111, produz como saída o vetor $[-0, 5784 \ -0, 5784]^T$.

Tabela 2: Vetores-códigos bidimensionais efetivamente armazenados na memória de referência do codificador.

i	w_{i1}	w_{i2}	Representação binária
1	0,5784	0,5784	000
2	0,1365	0,1365	001
3	0,1020	0,1020	010
4	0,0765	0,0765	011
5	0,0550	0,0550	100
6	0,0357	0,0357	101
7	0,0176	0,0176	110
8	0,0010	0,0010	111

4.1. Número de Operações do Método Proposto

O método de codificação proposto no presente trabalho será denotado a partir de agora por 1/2PCA, em virtude de o codificador utilizar apenas metade do dicionário previamente projetado com o algoritmo PCA.

Para codificar um determinado vetor de entrada, o algoritmo 1/2PCA testa, inicialmente, a média desse vetor. Para tanto, é necessário apenas avaliar a soma das componentes desse vetor:

$$\begin{cases} \text{soma}(\mathbf{x}) \geq 0 \Rightarrow \text{média}(\mathbf{x}) \geq 0 \\ \text{soma}(\mathbf{x}) < 0 \Rightarrow \text{média}(\mathbf{x}) < 0 \end{cases} \quad (3)$$

O teste da média requer, portanto, $K - 1$ adições (uma vez que a soma das componentes do vetor de entrada é dada por $\text{soma}(\mathbf{x}) = \sum_{j=1}^K x_j$) e 1 comparação (com 0).

Considerando a medida de distorção erro quadrático da Equação (2) e tendo em vista que o codificador realiza a busca pelo vizinho mais próximo em apenas $N/2$ vetores-códigos, observa-se que para codificar cada vetor de entrada são necessárias ainda $K(N/2)$ multiplicações, $K(N/2)$ subtrações, $(K - 1)(N/2)$ adições e $(N/2) - 1$ comparações.

O número total de adições do algoritmo de codificação 1/2PCA é, portanto, dado por $(K - 1) + (K - 1)(N/2)$, em que o primeiro termo da soma diz respeito ao “teste da média” e o segundo se refere ao processo de procura pelo vizinho mais próximo no dicionário com $N/2$ vetores-códigos. O número total de comparações, por sua vez, é igual a $1 + (N/2 - 1)$, em que o primeiro termo da soma se deve ao “teste da média” e o segundo termo é concernente às comparações de distância (distorção quadrática) do processo de busca pelo vizinho mais próximo.

A Tabela 3 apresenta um resumo do número total de operações envolvido no algoritmo de busca total (busca exaustiva pelo vizinho mais próximo, realizada de forma convencional em um dicionário com N vetores-códigos), denotado por BT, e no algoritmo 1/2PCA (em que apenas metade do dicionário, correspondente a $N/2$ vetores-códigos, é efetivamente utilizada pelo codificador).

Tabela 3: Número de operações requerido para codificar um vetor, ao serem utilizados os algoritmos BT e 1/2PCA, em função de K e N .

	N ^o de operações	
	BT	1/2PCA
×	KN	$KN/2$
-	KN	$KN/2$
+	$(K - 1)N$	$(K - 1)(1 + N/2)$
Comp.	$N - 1$	$N/2$

5. RESULTADOS

A Tabela 3 mostra que o algoritmo 1/2PCA apresenta uma economia de 50% em termos do número de multiplicações (bem como subtrações) com relação ao algoritmo BT. Convém mencionar que, de acordo com [9], o número de multiplicações é geralmente utilizado como critério de avaliação da complexidade computacional da etapa de codificação da QV.

Conforme se pode observar na Tabela 4, à medida que N aumenta o algoritmo 1/2PCA tende a levar a uma economia de 50% em termos do número de adições em relação ao algoritmo BT. Na Tabela 4 observa-se que para $K = 4$ e $N = 128$ o algoritmo 1/2PCA requer 195 adições, o que representa uma economia de 49,21% em relação às 384 adições requeridas pelo algoritmo BT.

No que diz respeito ao número de comparações, a Tabela 5 mostra que para $N = 32$ e $N = 64$ são obtidas, respectivamente, economias de 48,38% e 49,20% com a utilização do algoritmo 1/2PCA em lugar do algoritmo BT. Observa-se que a economia do número de comparações do algoritmo 1/2PCA com relação ao algoritmo BT tende a 50% à medida que N aumenta.

Tabela 4: Número de adições requerido para codificar um vetor, ao serem utilizados os algoritmos BT e 1/2PCA, para diversos valores de N fixada a dimensão $K = 4$. Entre parênteses está indicada a economia do algoritmo 1/2PCA com relação ao algoritmo BT.

N	N ^o de adições	
	BT	1/2PCA
16	48	27 (43,75%)
32	96	51 (46,87%)
64	192	99 (48,43%)
128	384	195 (49,21%)

Tabela 5: Número de comparações requerido para codificar um vetor, ao serem utilizados os algoritmos BT e 1/2PCA, para diversos valores de N . Entre parênteses está indicada a economia do algoritmo 1/2PCA com relação ao algoritmo BT.

N	N ^o de comparações	
	BT	1/2PCA
16	15	8 (46,67%)
32	31	16 (48,38%)
64	63	32 (49,20%)
128	127	64 (49,60%)

Foram realizadas simulações que tiveram como objetivo avaliar a queda de qualidade do sinal de voz reconstruído obtida ao se utilizar o algoritmo 1/2PCA em substituição ao algoritmo BT. A Tabela 6 apresenta valores de relação sinal-ruído segmental (SNRseg) [10, 11] do sinal de voz reconstruído para diversos valores de K e N . É possível observar que o algoritmo 1/2PCA pode eventualmente levar a uma pequena queda de SNRseg do sinal reconstruído quando comparado com o algoritmo BT. Para $K = 2$ e $N = 128$, por exemplo, obtém-se uma queda de 0,16 dB em termos de SNRseg do sinal reconstruído. As perdas de qualidade em termos de relação sinal-ruído segmental introduzidas pelo algoritmo 1/2PCA ocorrem quando o “teste da média” desse algoritmo leva um vetor de entrada a ser representado por um vetor-código do tipo $w_{i,PZ}$ quando de fato esse vetor de entrada deveria ser representado por um vetor-código

do tipo $w_{i,NZ}$ caso o algoritmo BT (busca total pelo vizinho mais próximo no dicionário completo, de N vetores-código) fosse utilizado. Também ocorrem quando o “teste da média” do algoritmo 1/2PCA leva um vetor de entrada x a ser representado por um vetor-código do tipo $w_{i,NZ}$ quando de fato x deveria ser representado por um vetor-código do tipo $w_{i,PZ}$ caso o algoritmo BT fosse utilizado.

Tabela 6: SNRseg (dB) do sinal de voz reconstruído, obtidas ao serem utilizados os algoritmos BT e 1/2PCA, para diversos valores de K e N .

K	N	SNRseg (dB)	
		BT	1/2PCA
2	8	3,76	3,76
2	16	8,86	8,86
2	32	11,01	10,84
2	64	12,35	12,35
2	128	13,41	13,25

A Figura 3 ilustra um caso em que o algoritmo 1/2PCA determina como versão quantizada do vetor de entrada x um vetor-código que difere do que deveria ser determinado caso o algoritmo BT fosse utilizado. Na figura, as coordenadas x_1 e x_2 representam, respectivamente, a primeira e a segunda componentes de qualquer vetor (como x , w_8 e w_9) definido no espaço euclidiano R^2 . Observa-se que o algoritmo 1/2PCA, em virtude de realizar o “teste da média” (por meio do qual verifica que o vetor de entrada trata-se de um vetor com média negativa) e, conseqüentemente, informar ao decodificador que o vetor-código a ser produzido é um vetor do tipo $w_{i,NZ}$, faz com que o vetor-código escolhido como versão quantizada de x seja o vetor w_9 . O algoritmo BT, por sua vez, que realiza uma busca no dicionário completo, determina que o vetor-código w_8 seja produzido como a versão quantizada de x . No exemplo da Figura 3, portanto, observa-se que o “teste da média” faz com que o algoritmo 1/2PCA leve um vetor de entrada a ser representado por um vetor-código do tipo $w_{i,NZ}$ (vetor w_9) quando de fato o vetor x deveria ser representado por um vetor-código do tipo $w_{i,PZ}$ (vetor w_8) caso o algoritmo BT fosse utilizado. Na figura, a distância entre os vetores x e w_8 é denotada por d_8 e a distância entre os vetores x e w_9 é denotada por d_9 . Como $d_8 < d_9$ um maior erro de quantização é obtido com a utilização de w_9 . Em decorrência de poder eventualmente levar a um maior erro de quantização quando comparado ao obtido com o algoritmo BT, o algoritmo 1/2PCA pode levar a sinais reconstruídos com valores de SNRseg inferiores aos obtidos com uso do algoritmo BT.

É importante ressaltar que o fato de d_8 ser menor que d_9 não necessariamente implica que w_8 é mais adequado que w_9 para representação de x tendo em vista aspectos de qualidade subjetiva do sinal de voz reconstruído. As-

sim, o algoritmo 1/2PCA, que apresenta uma semelhança com a técnica de QV classificada [12] em virtude de realizar um pré-classificação de acordo com a média do vetor a ser codificado, pode levar a sinais reconstruídos com qualidade subjetiva superior à obtida com o uso do algoritmo BT. Nesse contexto, testes de escuta informais mostraram que os sinais obtidos com algoritmo 1/2PCA apresentam uma qualidade ligeiramente superior à apresentada pelos sinais obtidos com o algoritmo BT.

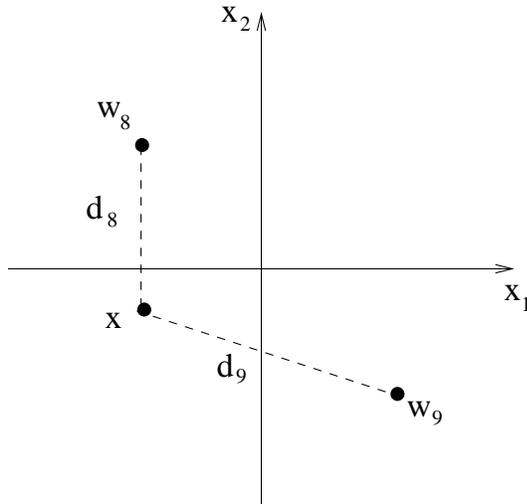


Figura 3: Exemplo em que os algoritmos 1/2PCA e BT levam a representações diferentes para um dado vetor de entrada x .

6. CONCLUSÃO

Neste trabalho a simetria existente nos dicionários PCA foi utilizada para reduzir os requisitos de capacidade de memória exigida para armazenar os vetores-códigos bem como para reduzir significativamente a complexidade computacional (número de operações) envolvida na fase de codificação da quantização vetorial.

Com o método de codificação proposto, o codificador do sistema de comunicação baseado em QV realiza uma busca pelo vizinho mais próximo em apenas metade dos vetores-códigos do dicionário. Ao se utilizar como medida de distorção a distância quadrática, o método proposto leva a uma economia de 50% do número de multiplicações e subtrações quando comparado com o algoritmo convencional de busca total (BT). Além disso, ao ser considerado um dicionário de dimensão K e tamanho N , o método proposto requer $(K - 1)(1 + N/2)$ adições e $N/2$ comparações para codificar um vetor de entrada, enquanto o algoritmo BT requer $(K - 1)N$ adições e $N - 1$ comparações. Observou-se que o método de codificação apresentado neste trabalho pode eventualmente introduzir uma pequena

queda de relação sinal-ruído segmental dos sinais reconstruídos, quando comparado ao valor de SNRseg obtido com a utilização do algoritmo BT. Sob o ponto de vista da qualidade subjetiva dos sinais de voz reconstruídos, testes de escuta informais mostraram que os sinais obtidos com algoritmo de codificação proposto apresentam uma qualidade ligeiramente superior à apresentada pelos sinais obtidos com o algoritmo BT.

Finalmente, é oportuno mencionar que o método de codificação proposto, embora tenha sido aplicado aos dicionários PCA, pode ser extensivo a quaisquer dicionários que apresentem simetria entre os vetores-códigos.

7. AGRADECIMENTOS

Os autores expressam os agradecimentos à CAPES e ao CNPq pelo apoio financeiro ao trabalho.

8. REFERÊNCIAS

- [1] Madeiro, F., Vilar, R. M., Aguiar Neto, B. G., and Alencar, M. S. An Algorithm for Voice Waveform VQ Codebook Design Based on PCA. *Revista da Sociedade Brasileira de Telecomunicações*, Vol. 15, No. 2, pp. 59–69, Dezembro, 2000.
- [2] Gersho, A. and Gray, R. M. *Vector Quantization and Signal Compression*. Kluwer Academic Publishers, Boston, MA, 1992.
- [3] Gray, R. M. Vector Quantization. *IEEE ASSP Magazine*, pp. 4–29, April 1984.
- [4] Madeiro, F. Projeto e Avaliação de Dicionários para Quantização Vetorial de Voz e Imagem. *Proposta de Tese, Universidade Federal da Paraíba, Departamento de Engenharia Elétrica*, 2001.
- [5] Linde, Y., Buzo, A., and Gray, R. M. An Algorithm for Vector Quantizer Design. *IEEE Transactions on Communications*, Vol. COM - 28, No. 1, pp. 84–95, January 1980.
- [6] Kohonen, T. *Self-Organization and Associative Memory (3rd ed)*. Springer-Verlag, Berlin, 1989.
- [7] Kohonen, T. The Self-Organizing Map. *Proceedings of the IEEE*, Vol. 78, No. 9, pp. 1464–1480, September 1990.
- [8] Madeiro, F., Vilar, R. M., Fechine, J. M., and Aguiar Neto, B. G. A Self-Organizing Algorithm for Vector Quantizer Design Applied to Signal Processing. *International Journal of Neural Systems*, Vol. 9, No. 3, pp. 219–226, June 1999.
- [9] Bei, C.-D. and Gray, R. M. An Improvement of the Minimum Distortion Encoding Algorithm for Vector Quantization. *IEEE Transactions on Communications*, Vol. COM-33, No. 10, pp. 1132–1133, October 1985.
- [10] Deller Jr., J. R., Proakis, J. G., and Hansen, J. H. L. *Discrete-time Processing of Speech Signals*. Macmillan Publishing Co., 1993.
- [11] Jayant, N. S. and Noll, P. *Digital Coding of Waveforms*. Prentice-Hall, Englewood Cliffs, NJ, 1984.
- [12] Ramamurthi, B. and Gersho, A. Classified Vector Quantization of Images. *IEEE Transactions on Communications*, Vol. COM-34, No. 11, pp. 1105–1115, November 1986.