

Análise de uma Estratégia de Redução Computacional para o Algoritmo de Matching Pursuit com Dicionários Diádicos

Gabriel Almeida de Oliveira, Michel Pompeu Tcheou e Lisandro Lovisoló

Resumo— Este artigo apresenta uma análise de uma estratégia de redução de complexidade computacional do algoritmo de Matching Pursuit (MP), com base em um dicionário redundante com discretização diádica. A estratégia calcula os produtos internos entre o sinal e os átomos do dicionário, para diferentes frequências, fixando-se a escala e o deslocamento no tempo, através da Transformada Rápida de Fourier. São apresentados tempos computacionais de modo a comparar os desempenhos do MP clássico e do MP rápido. As implementações foram realizadas em linguagem C++ e executadas em ambiente Linux.

Palavras-Chave— Processamento de sinais, decomposições atômicas, Matching Pursuit.

Abstract— This article presents an analysis of a computational complexity reduction strategy of Matching Pursuit (MP) algorithm, based on a dyadic redundant dictionary. The strategy compute inner products between the signal and dictionary atoms, for different frequencies, fixing scale and time shift, by using Fast Fourier Transform (FFT). Computation time are presented so as to comparing performance of the classic and fast MP versions. The implementations were devised in C++ programming language and Linux environment.

Keywords— Signal processing, atomic decompositions, Matching Pursuit.

I. INTRODUÇÃO

O algoritmo Matching Pursuit (MP) proposto em [1] realiza aproximações sucessivas de um sinal \mathbf{x} , de forma adaptativa e iterativa, buscando em um dicionário redundante a melhor aproximação a cada iteração. Inicialmente, tem-se o resíduo $\mathbf{r}_x^0 = \mathbf{x}$. A cada iteração i :

- 1) Calculam-se as correlações entre o resíduo \mathbf{r}_x^{i-1} e cada átomo \mathbf{g}_γ do dicionário redundante D , utilizando-se produtos internos:

$$\forall \mathbf{g}_\gamma \in D, \langle \mathbf{r}_x^{i-1}, \mathbf{g}_\gamma \rangle; \quad (1)$$

- 2) Busca-se o átomo mais correlacionado com o resíduo, procurando-se o produto interno máximo:

$$\hat{\mathbf{g}}_{\gamma(i)} = \operatorname{argmáx}_{\mathbf{g}_\gamma \in D} |\langle \mathbf{r}_x^{i-1}, \mathbf{g}_\gamma \rangle|; \quad (2)$$

- 3) Subtrai-se o átomo ponderado $\alpha_i \mathbf{g}_{\gamma(i)}$ do resíduo \mathbf{r}_x^{i-1} , onde $\alpha_i = \langle \mathbf{r}_x^{i-1}, \hat{\mathbf{g}}_{\gamma(i)} \rangle$:

$$\mathbf{r}_x^i = \mathbf{r}_x^{i-1} - \alpha_i \hat{\mathbf{g}}_{\gamma(i)}; \quad (3)$$

- 4) Se o critério de parada for satisfeito, termina-se a decomposição; em caso contrário, faz-se $i = i + 1$ e retorna-se ao passo 1.

Ao fim da decomposição, obtém-se a seguinte aproximação em M termos do sinal \mathbf{x} :

$$\hat{\mathbf{x}} = \sum_{i=1}^M \alpha_i \hat{\mathbf{g}}_{\gamma(i)}. \quad (4)$$

Para um dicionário de dimensão N , observa-se que cada produto interno $\langle \mathbf{r}_x^{i-1}, \mathbf{g}_\gamma \rangle$ requer N multiplicações e $N - 1$

adições, e em cada iteração calculam-se $\#_D \geq N$ ($\#_D$ corresponde à cardinalidade do dicionário) produtos internos. Portanto, o custo do cálculo dos produtos internos no passo 1 atinge $\mathcal{O}(N^2)$ a cada iteração, tornando o uso do MP inviável para sinais de dimensão grande.

No passo 2, realiza-se, normalmente, uma varredura por todas as correlações obtidas, o que corresponde a uma complexidade computacional de $\mathcal{O}(\#_D)$ por iteração. Dado que são realizadas M iterações, o custo computacional total $\mathcal{O}(M(N^2 + \#_D))$ de implementações clássicas de MP ultrapassa $\mathcal{O}(N^2)$, tornando o uso do MP ainda mais impraticável computacionalmente.

II. PRODUTO INTERNO ENTRE O RESÍDUO E O ÁTOMO

Em [2], é proposto um método rápido para o cálculo do produto interno entre o sinal e um átomo real do dicionário apresentando fase ótima. Os elementos de um dicionário redundante podem ser gerados escalonando-se, trasladando-se e modulando-se uma função-janela $g(n)$ no tempo discreto. No caso de um dicionário de átomos reais, cada átomo $\mathbf{g}_{\gamma,\phi}(n)$ é definido como

$$\mathbf{g}_{\gamma,\phi}(n) = \frac{g\left(\frac{n-\tau}{s}\right) \cos(\xi n + \phi)}{\|g\left(\frac{n-\tau}{s}\right) \cos(\xi n + \phi)\|}, \quad (5)$$

em que $\gamma = (s, \tau, \xi)$, s é a escala, τ é o deslocamento, ξ é a frequência e ϕ representa a fase. O produto interno $\langle \mathbf{x}, \mathbf{g}_{\gamma,\phi} \rangle$ e sua fase ótima ϕ_0 correspondente, onde $\max_{\phi \in [0, 2\pi]} |\langle \mathbf{x}, \mathbf{g}_{\gamma,\phi} \rangle|$, são definidos por

- 1) Se $\xi = 0$, então $\phi_0 = 0$ e $\langle \mathbf{x}, \mathbf{g}_{\gamma,\phi} \rangle = \frac{\langle \mathbf{x}, \mathbf{P} \rangle}{\|\mathbf{P}\|}$.
- 2) Se $\xi \neq 0$ e $a_1 = 0$, então $\phi_0 = \frac{\pi}{2}$ e $\langle \mathbf{x}, \mathbf{g}_{\gamma,\phi} \rangle = -\frac{\langle \mathbf{x}, \mathbf{Q} \rangle}{\|\mathbf{Q}\|}$.
- 3) Se $\xi \neq 0$ e $a_1 \neq 0$, então $\phi_0 = \operatorname{arctg}\left(-\frac{b_1}{a_1}\right)$ e $\langle \mathbf{x}, \mathbf{g}_{\gamma,\phi} \rangle = \frac{\langle \mathbf{x}, \mathbf{P} \rangle a_1 + \langle \mathbf{x}, \mathbf{Q} \rangle b_1}{\|\mathbf{P} a_1 + \mathbf{Q} b_1\|}$.

onde

$$a_1 = \langle \mathbf{x}, \mathbf{P} \rangle \|\mathbf{Q}\|^2 - \langle \mathbf{x}, \mathbf{Q} \rangle \langle \mathbf{P}, \mathbf{Q} \rangle, \quad (6)$$

$$b_1 = \langle \mathbf{x}, \mathbf{Q} \rangle \|\mathbf{P}\|^2 - \langle \mathbf{x}, \mathbf{P} \rangle \langle \mathbf{P}, \mathbf{Q} \rangle, \quad (7)$$

e os vetores \mathbf{P} e \mathbf{Q} são definidos como

$$\mathbf{P}(n) = g\left(\frac{n-\tau}{s}\right) \cos(\xi n), \quad \mathbf{Q}(n) = g\left(\frac{n-\tau}{s}\right) \operatorname{sen}(\xi n). \quad (8)$$

Observa-se, portanto, que para se obter a projeção $\langle \mathbf{x}, \mathbf{g}_{\gamma,\phi_0} \rangle$ do vetor sinal sobre o átomo de fase ótima são necessários cinco fatores: $\langle \mathbf{x}, \mathbf{P} \rangle$, $\langle \mathbf{x}, \mathbf{Q} \rangle$, $\langle \mathbf{P}, \mathbf{P} \rangle$ (i.e. $\|\mathbf{P}\|^2$), $\langle \mathbf{Q}, \mathbf{Q} \rangle$ (i.e. $\|\mathbf{Q}\|^2$), $\langle \mathbf{Q}, \mathbf{P} \rangle$.

III. ESTRATÉGIA DE REDUÇÃO DE COMPLEXIDADE

É possível calcular N produtos internos através da transformada rápida de Fourier (FFT - *Fast Fourier Transform*) para N valores de frequência ξ , tornando, assim, o passo 1 do algoritmo de MP mais célere. Para que se aproveite ao máximo o potencial computacional da FFT é necessário que o comprimento N do sinal seja potência de 2. O parâmetro de frequência deve ser discretizado da seguinte forma: $\xi = \frac{2\pi k}{N}$, $k = 0, 1, \dots, N - 1$. Os valores de escala s e de deslocamento

G. A. de Oliveira, M. P. Tcheou, L. Lovisoló, Faculdade de Engenharia, Universidade do Estado do Rio de Janeiro, Rio de Janeiro-RJ, Brasil, E-mails: gabrielalmeida140395@gmail.com.br, mtcheou@uerj.br, lisandro@uerj.br.

τ podem ser definidos livremente. A metodologia de cálculo rápido dos produtos internos utilizando FFT, proposta em [2], é apresentada a seguir. Inicialmente, para uma dada escala s e deslocamento τ , definem-se

$$P_k(n) = g\left(\frac{n-\tau}{s}\right) \cos\left(\frac{2\pi nk}{N}\right), \quad (9)$$

$$Q_k(n) = g\left(\frac{n-\tau}{s}\right) \sin\left(\frac{2\pi nk}{N}\right). \quad (10)$$

Fazendo-se $\gamma = (s, \tau, 2\pi k/N)$, para cada valor de $k = 0, 1, \dots, N-1$, utiliza-se a FFT para calcular as grandezas

$$\langle \mathbf{x}, \mathbf{P}_k \rangle, \langle \mathbf{x}, \mathbf{Q}_k \rangle, \langle \mathbf{P}_k, \mathbf{P}_k \rangle, \langle \mathbf{Q}_k, \mathbf{P}_k \rangle \text{ e } \langle \mathbf{Q}_k, \mathbf{Q}_k \rangle,$$

que são usadas no cálculo da projeção $\langle \mathbf{x}, \mathbf{g}_{\gamma, \phi} \rangle$. Seja a transformada discreta de Fourier (DFT) $Y(k)$ de uma sequência $y(j)$ de comprimento N

$$Y(k) = \sum_{j=0}^N y(j) \cos\left(\frac{2\pi jk}{N}\right) - i \sum_{j=0}^N y(j) \sin\left(\frac{2\pi jk}{N}\right). \quad (11)$$

Ao se realizar a transformada de Fourier da sequência $y_1(j) = x(j)g\left(\frac{j-\tau}{s}\right)$, obtêm-se para $k = 0, 1, \dots, N-1$ os produtos internos

$$\langle \mathbf{x}, \mathbf{P}_k \rangle = \text{Re}\{Y_1(k)\}, \langle \mathbf{x}, \mathbf{Q}_k \rangle = -\text{Im}\{Y_1(k)\}. \quad (12)$$

Agora, considerando como entrada da transformada de Fourier a sequência $y_2(j) = g^2\left(\frac{j-\tau}{s}\right)$, obtêm-se

$$\langle \mathbf{P}_k, \mathbf{P}_k \rangle = \frac{1}{2} (\text{Re}\{Y_2(0)\} + \text{Re}\{Y_2(2k)\}), \quad (13)$$

$$\langle \mathbf{Q}_k, \mathbf{Q}_k \rangle = \frac{1}{2} (\text{Re}\{Y_2(0)\} - \text{Re}\{Y_2(2k)\}), \quad (14)$$

$$\langle \mathbf{Q}_k, \mathbf{P}_k \rangle = -\frac{1}{2} \text{Im}\{Y_2(2k)\}, \quad (15)$$

para $k = 0, 1, \dots, \frac{N}{2} - 1$, e

$$\langle \mathbf{P}_k, \mathbf{P}_k \rangle = \left\langle \mathbf{P}_{k-\frac{N}{2}}, \mathbf{P}_{k-\frac{N}{2}} \right\rangle, \quad (16)$$

$$\langle \mathbf{Q}_k, \mathbf{Q}_k \rangle = \left\langle \mathbf{Q}_{k-\frac{N}{2}}, \mathbf{Q}_{k-\frac{N}{2}} \right\rangle, \quad (17)$$

$$\langle \mathbf{Q}_k, \mathbf{P}_k \rangle = \left\langle \mathbf{Q}_{k-\frac{N}{2}}, \mathbf{P}_{k-\frac{N}{2}} \right\rangle, \quad (18)$$

para $k = \frac{N}{2}, \dots, N-1$.

O algoritmo da implementação do MP utilizando esta estratégia apresenta uma complexidade $\mathcal{O}(2N \log_2 N)$. Sua grande vantagem é obter os produtos internos entre o resíduo e N átomos através do uso de duas FFTs. Contudo, sua desvantagem é que o espaçamento entre as frequências do dicionário, nesse caso, é obrigatoriamente linear.

IV. RESULTADOS EXPERIMENTAIS

Para obter os tempos de execução dos algoritmos, foi utilizado um computador com o processador Intel(R) Core(TM) i7-3770 CPU @ 3,4 GHz, com 8 Gb de memória RAM, e sistema operacional Linux. O sinal de áudio processado corresponde a um sinal de piano associado à nota musical A3, como ilustra a Figura 1. Este sinal possui a duração de 1 segundo e taxa de amostragem 44,1 kHz, portanto são 44100 amostras de comprimento.

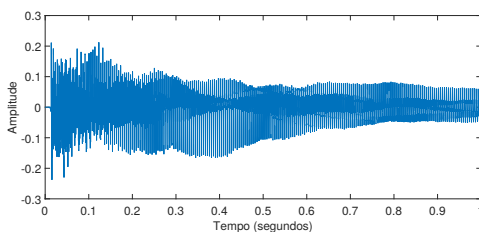


Fig. 1. Sinal de áudio correspondente à nota A3 de um piano.

Neste experimento, adota-se o dicionário de exponenciais com discretização diádica, ou seja, os átomos são definidos da seguinte forma:

$$g_{\gamma_d}(n) = g_j(n) \cos(nk\pi 2^{1-j} + \phi) \cdot [u(n-n^s) - u(n-n^e)],$$

$$n = \{0, \dots, N-1\} \quad (19)$$

$$g_j(n) = \begin{cases} \delta(j), & j = 0 \\ K_{\gamma_d} e^{-m_f 2^{-j}(n-p2^j)}, & m_f = 1 \text{ e } j \in [1, L) \\ K_{\gamma_d} e^{-m_f 2^{-j}(p2^j-n)}, & m_f = -1 \text{ e } j \in [1, L) \\ \frac{1}{\sqrt{N}}, & j = L \end{cases}. \quad (20)$$

O dicionário possui $L = \log_2(N)$ escalas e os intervalos dos parâmetros que definem os elementos são $j \in [0, L]$, $p \in [0, N2^{-j}]$ e $k \in [0, 2^j]$. A decomposição é feita em blocos de 512 amostras com janela retangular e sem sobreposição. O critério de parada usado foi o número máximo de iterações igual a 512. Os algoritmos usados na decomposição foram desenvolvidos em linguagem de programação C++.

As complexidades e os resultados experimentais obtidos para as versões clássica e rápida do MP são apresentados na Tab. I. Conforme os resultados obtidos, pode-se observar que a versão rápida do MP é aproximadamente 25 vezes mais rápida que a versão clássica, quando utilizado um dicionário de exponenciais com discretização diádica e dimensão 512. A partir do desvio padrão é possível verificar que a dispersão não é significativa quando comparada à média em ambos os casos.

TABELA I

COMPLEXIDADES E TEMPOS COMPUTACIONAIS MEDIDOS.

	MP clássico	MP rápido [2]
Complexidade	$\mathcal{O}(N^2)$	$\mathcal{O}(2N \log_2 N)$
Tempo médio por bloco (seg.)	242,02	9,6312
Desvio padrão por bloco	0,86032	0,44458
Tempo médio por iteração (seg.)	0,47269	0,01881

V. CONCLUSÕES

Neste trabalho, avaliou-se o desempenho de uma estratégia de aceleração do algoritmo de Matching Pursuit (MP), utilizando um dicionário de exponenciais com discretização diádica e dimensão 512. Esta estratégia se baseia basicamente no uso da Transformada Rápida de Fourier para o cálculo dos produtos internos. Através dos resultados obtidos, pode ser verificada uma redução do tempo de execução da decomposição, tornando-se mais significativa com o aumento do dicionário. Destaca-se que, ao aumentar a dimensão do dicionário, o tempo de execução não aumenta linearmente. Isto pode ser concluído comparando-se as complexidades das versões clássica e rápida do MP.

REFERÊNCIAS

- [1] S. G. Mallat e Z. Zhang, "Matching Pursuits with Time-Frequency Dictionaries," *IEEE Transactions on Signal Processing*, v. 41, n. 12, pp. 3397-3415, Dezembro 1993.
- [2] S. E. Ferrando, L. A. Kolas e Natasha Kovacevic, "Algorithm 820: A Flexible Implementation of Matching Pursuit for Gabor Functions on the Interval," *ACM Transactions on Mathematical Software*, v. 28, n. 3, pp. 337-353, Setembro 2002.