SEPARAÇÃO CEGA DE FONTES: APLICANDO UM ALGORITMO PARA DECOMPOSIÇÃO EM VECTORES PRÓPRIOS

Nuno Ferreira[1], Ana Maria Tomé

Resumo – **Neste trabalho é apresentado um algoritmo iterativo para separação cega de fontes (BSS – *Blind Source Separation*). O algoritmo é baseado na decomposição em valores e vectores próprios de um par de matrizes. Para o estudo e demonstração do algoritmo foi desenvolvido, em *Matlab*, um pacote de software. Finalmente, são apresentados resultados experimentais, com misturas de sinais de áudio, calculando o par de matrizes à entrada e à saída de um filtro linear de resposta impulsional finita.**

Abstract – **This work presents an iterative algorithm for blind source separation. The algorithm is based on the generalized eigendecomposition of a matrix pencil. A *Matlab* package of software was developed to evaluate the algorithm performance. Some experimental results, using artificially mixed audio signals and a matrix pencil computed at the input and at the output of a linear finite impulse response, are also presented.**

I. INTRODUÇÃO

O modelo matemático utilizado na formulação de um problema de separação dos sinais misturados pode ser descrito pela expressão,

$$y(t) = Ms(t) \tag{1}$$

Nesta expressão o vector dos sinais misturados, ***y(t)***, é conhecido, enquanto que a matriz de mistura (*M*) e o vector de sinais fonte ***s(t)*** são desconhecidos. A separação dos sinais misturados, é realizada recorrendo a métodos baseados na decomposição em valores próprios [1,2,3].

Este método consiste na diagonalização de um par de matrizes calculadas nos sinais misturados. A decomposição em valores/vectores próprios do par de matrizes *(B,A)* é dada pela seguinte expressão,

$$AE = BED \tag{2}$$

A matriz dos vectores próprios (*E*) será uma estimativa da matriz de separação se a matriz dos valores próprios (*D*) tiver valores únicos na diagonal principal [3].

Existem actualmente vários métodos para determinação das matrizes de correlação *(B,A)*. Neste trabalho foram aplicados dois métodos distintos: primeiro método consiste na utilização de matrizes de correlação determinadas a partir dos sinais misturados (*B*) e de uma versão atrasada dos mesmos sinais misturados (*A*) [1]; e o segundo método na utilização de matrizes de correlação determinadas a partir de uma filtragem dos sinais misturados (FIR) [2,3] – tal como mostra a figura 1.

Neste trabalho descrevemos um algoritmo iterativo, i.e., com actualização dos valores/vectores próprios amostra a amostra. Para além disso, foi ainda desenvolvido um pacote de software em *Matlab* com o objectivo de estudar o desempenho do algoritmo com misturas artifíciais de sinais de áudio.

II. ALGORITMO ITERATIVO

O cálculo de valores próprios de um par de matrizes *(B,A)*, se *B* for simétrica definida positiva pode ser efectuado em dois passos [4]. Em qualquer um dos passos é calculada a decomposição dos valores próprios (e respectivos vectores próprios) de uma matriz: no 1º passo a decomposição de *B*, no 2º passo a decomposição de uma transformação linear de *A*.

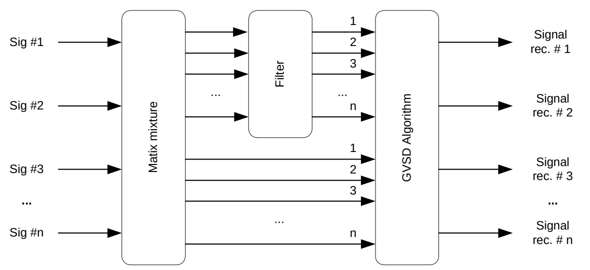

Fig. 1 – Modelo base utilizado para a estimativa de parâmetros para recuperação de sinais misturados (utilizando um filtro no calculo das matrizes de correlação).

Sendo *B* uma matriz real simétrica definida positiva ($B = B^T$), a sua decomposição em valores próprios (Δ) e vectores próprios (*S*) permite escrever a equação seguinte:

[1] Este projecto é financiado pela "Fundação para a Ciência e a Tecnologia (FCT)" – ICAVIP POSI / 34246/SRI/2000.

$$B = S\Delta S^T \tag{3}$$

Como S é uma matriz unitária, i.e. $S^TS = I$, a matriz B, é descrita pela equação seguinte:

$$B = S\Delta^{1/2}S^T S\Delta^{1/2}S^T = WW \tag{4}$$

Considerando $S\Delta^{1/2}S^T = W$, temos que a inversa da matriz W é dada por,

$$W^{-1} = S\Delta^{-1/2}S^T \tag{5}$$

Manipulando a expressão (2), com as equações (4) e (5) e considerando $Z = WE$, obtém-se a equação seguinte:

$$W^{-1}AW^{-1}Z = ZD \tag{6}$$

A equação anterior formula a decomposição em valores próprios (D) e vectores próprios (Z) da matriz $C = W^{-1}AW^{-1}$.

Os valores próprios da equação (6) são os valores próprios da equação (1). Para calcular os respectivos vectores próprios, basta resolver a seguinte equação,

$$E = W^{-1}Z \tag{7}$$

No algoritmo iterativo proposto, a decomposição em valores e vectores próprios de B e C são incluidas no ciclo principal. Para isso, o cálculo da matriz de transformação W^{-1} é baseada na decomposição espectral,

$$W^{-1} = \sum_i \frac{1}{\sqrt{\delta_i}} \cdot s_i^T \cdot s_i \tag{8}$$

onde cada par (δ_i, s_i) será incluído na soma da equação anterior se for considerado uma estimativa válida para os valores próprios e vectores próprios de B, i.e., se o erro quadrático,

$$e(i) = \left(Bs_i - \delta_i s_i\right)^T \cdot \left(Bs_i - \delta_i s_i\right) \tag{9}$$

tiver um valor inferior a um limiar pré-definido.

Por outro lado a decomposição em valores e vectores próprios de B e C é executada iterativamente aplicando:

- Método das Potências que é um método sistemático de procura do valor próprio máximo e respectivo vector próprio de uma matriz simétrica definida positiva [5].
- Método da Deflação, i.e., calculando uma nova matriz $B^j = B - \delta_i s_i^T s_i$ após o cálculo de um par (δ_i, s_i). À nova matriz será aplicado o método da potência [5].

A tabela (1) apresenta os passos principais do algoritmo desenvolvido.

III. DEMONSTRADOR

Foi desenvolvido em *Matlab* um demonstrador capaz de criar misturas artificiais (a partir de sinais de áudio armazenados em ficheiros *.*wav*), estimar a matriz de separação da mistura e uma forma de avaliar os resultados obtidos. Este demonstrador pode ser dividido em dois blocos distintos: primeiro bloco, cuja função principal consiste na recuperação dos sinais misturados e um segundo bloco, que permite a realização do tratamento dos dados obtidos do bloco anterior.

No primeiro bloco é permitida a configuração da simulação, onde é possível escolher-se os ficheiros .*wav* (a ler de um directório), o número de misturas, o método para a determinação do par de matrizes *(B,A)*, bem como guardar dados importantes da simulação para uma futura análise. Se o método escolhido para a determinação das matrizes de correlação for o processo de filtragem dos sinais misturados, é possível a escolha do ficheiro *.*dat* com os coeficientes do filtro a utilizar na simulação.

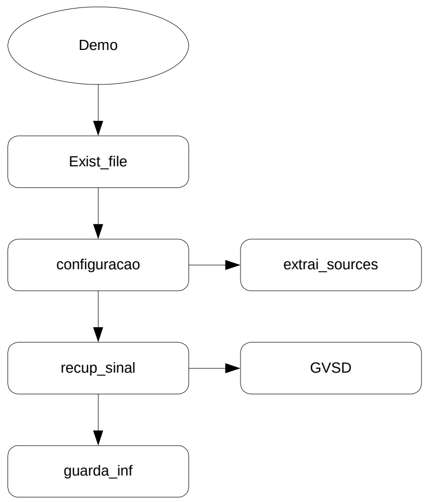

Fig. 2 – Diagrama de blocos do demonstrador.

O diagrama de blocos seguinte mostra a sequência de operação de uma sessão de demosntração, onde cada bloco corresponde a uma função *Matlab*.

- Função *Exist_file* – verifica a existência de ficheiros de dados de simulações anteriores no directório de trabalho. Se existirem ficheiros de dados nesse directório, é criado um novo directório e copiado para lá esses ficheiros de dados.
- Função *configuracao* – permite efectuar a configuração de diversos parâmetros da simulação que se pretende implementar, ou seja, número de misturas a utilizar, possibilidade de se adicionar ruído aos sinais misturados, método a

utilizar para se determinar o par de matrizes de correlação. A função *configuracao* invoca uma nova função, *extrai_sources*, que permite extrair e validar os sinais *.wav* de um directório.

- Função *recup_sinal* – depois de se configurar a simulação pretendida, passa-se à fase de recuperação dos sinais misturados. Esta função, invoca a função *GVSD* onde é aplicado o algoritmo iterativo descrito anteriormente, responsável pela determinação da matriz de separação dos sinais misturados. A função *GVSD* tem quatro parâmetros de entrada: sinais misturados, sinais misturados filtrados\atrasados, o intervalo para se guardar os valores\vectores próprios em ficheiro e o método utilizado para a determinação do par de matrizes de correlação; e quatro parâmetros de saída: vectores próprios, valores próprios e as duas matrizes de correlação do último segmento.
- Função *guarda_inf* – finalmente, são guardadas em ficheiro informações úteis da simulação efectuda, como sejam o nº de amostras, o nº de sinais, a matriz de mistura e intervalo para se guardar informação. Esta informação guardada poderá ser utilizada para futuro tratamento de dados.

A segunda parte permite realizar um tratamento dos dados que foram guardados ao longo da simulação. A função *trat_dados* é composta por várias funções e permite ao utilizador avaliar os resultados obtidos para a recuperação dos sinais. Esta função disponibiliza um gráfico onde se pode visualizar a evolução de cada um dos valores próprios ao longo da simulação, ouvir os sinais recuperados bem como determinar os coeficientes de correlação entre os sinais de entrada e os sinais recuperados à saída do sistema.

IV. RESULTADOS

Foi realizado um estudo experimental utilizando misturas de quatro sons *wave*, amostrados a 8 KHz e com 50000 amostras cada: uma ambulância, um violino, piano e voz e música rock [6]. Os sinais foram misturados aleatóriamente.

Para a determinação das matrizes de correlação *(B,A)*, foi utilizado um filtro FIR. A figura (3) mostra a resposta em frequência do filtro utilizado. O filtro foi desenhado (com o comando *firls*) de modo a modificar de modo quase aleatório o conteúdo de frequêncio dos sinais. O par de matrizes de correlação são determinadas usando as matrizes dos sinais misturados à entrada e saída do filtro.

Os parâmetros resultantes da decomposição efectuada, valores e vectores próprios, são guardados num ficheiro em intervalos de 500 amostras. Na figura (4) podemos visualizar a evolução dos valores próprios ao longo do segmento.

A avaliação do estudo experimental realizado, foi efectuada através do cálculo do coeficiente de correlação entre os sinais da fonte (***s***) e os sinais recuperados (***z***).

```
% n --> nº de sources
% N --> nº de amostras
for k = 1, 2, ..., N
 % Decomposição valores\vectores próprios para 1ª matriz
 % Matrizes B e A actualizadas amostra a amostra
 B^1 = B
 % Estimativa valores/vectores próprios da matriz B (δ(j),s(j))

 for j = 1:n
   s(j)=B^j*s(j)
   s(j)=s(j)/||s(j)||
   δ(j)=(s(j))^T*B^j*s(j)
   e(j)=(B*s(j)-δ(j)*s(j))^T*(B*s(j)-δ(j)*s(j))
   if j ≠ n
     B^(j+1)=B^j–s(j)*δ(j)*(s(j))^T
   end
 end

 % Cálculo da matriz de transformação
 W^-1=0
 for j = 1:n
   if (δ(j) > 10^-4 & e_1(j) < tol)
     W^-1=W^-1+s(j)*(s(j))^T/sqrt(δ(j))
   end
 end

 % Transformação da 2ª Matriz
 C=W^-1*A*W^-1
 % Estimativa valores/vectores próprios da matriz C (d(j),z(j))
 for j = 1:n
   z(j)=C^j*z(j)
   aux=(z(j))^T*z(j)
   if aux ~= 0
     z(j)=z(j)/||z(j)||
     d(j)=(z(j))^T*C^j*z(j)
   else
     d(j)=0
   end
   if j ≠ n
     C^(j+1)=C^j–z(j)*d(j)*(z(j))^T
   end
 end
end
% Cálculo dos vectores próprios de (B,A)
E=W^-1Z
```

Tab. 1 – Agoritmo Iterativo.

Verifica-se que a qualidade de sinais recuperados varia ao longo do segmento de dados.

Da figura (4), observa-se que na zona das 16500 (k = 33) e 27000 (k = 54) iterações, dois dos quatro valores próprios estão muito próximos um do outro. Este facto vai levar a que dois sinais à saída do sistema não sejam recuperados correctamente. Temos dois sinais misturados à saída do sistema (ver tabelas 2 e 3). Por exemplo, o sinal recuperado z2 tem coeficientes de correlação elevados com os sinais fonte s1 e s2. O mesmo acontence com o sinal recuperado z3. Por sua vez, os sinais recuperados z1 e z4 apresentam coeficientes de correlação elevados apenas com um dos sinais fonte (ver tabela 2). Na tabela (3), ainda se verifica a mistura de dois sinais à saída do sistema, no entanto os coeficientes de correlação não são tão elevados.

Os sinais originais são recuperados com a matriz dos vectores próprios calculada após 40000 iterações. De facto, a partir das 40000 iterações (k = 80) observa-se que os valores próprios se mantêm significativamente afastados, resultando assim, na recuperação dos sinais originais (ver tabela (4)).

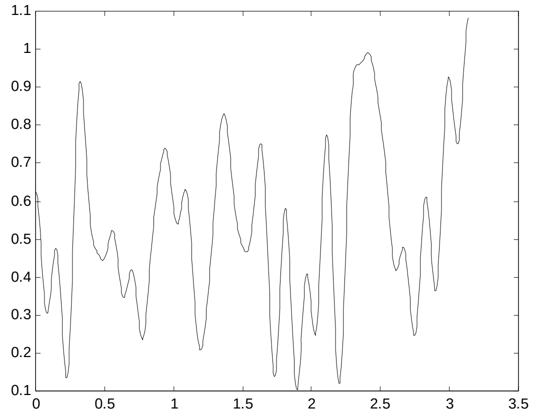

Fig. 3 – Resposta em frequência (módulo) para um filtro FIR com 100 coeficientes (—).

	s1	s2	s3	s4
z1	0.0061	0.0015	0.0023	0.9999
z2	0.7978	0.6020	0.0159	0.0096
z3	0.5563	0.8311	0.0229	0.0042
z4	0.0139	0.0227	0.9996	0.0006

Tab. 2 - Coeficiente de correlação entre os sinais originais e os sinais recuperados com parâmetros estimandos na iteração 16500.

	s1	s2	s3	s4
z1	0.0003	0.0008	0.0019	0.9999
z2	0.0450	0.9988	0.0212	0.5566
z3	0.9575	0.0585	0.2872	0.0018
z4	0.2461	0.0043	0.9679	0.0009

Tab. 3 - Coeficiente de correlação entre os sinais originais e os sinais recuperados com parâmetros estimandos na iteração 27000.

	s1	s2	s3	s4
z1	0.0007	0.0027	0.0003	0.9999
z2	0.0511	0.9986	0.0006	0.0041
z3	0.9979	0.0644	0.0081	0.0025
z4	0.0010	0.0047	0.9999	0.0016

Tab. 4 - Coeficiente de correlação entre os sinais originais e os sinais recuperados com parâmetros estimandos na iteração 40000.

Este estudo experimental foi efectuado com outros filtros e outras matrizes de mistura (M), tendo sido obtidos resultados da mesma natureza. Onde a evolução dos valores próprios indica a metodologia para a escolha da matriz de separação.

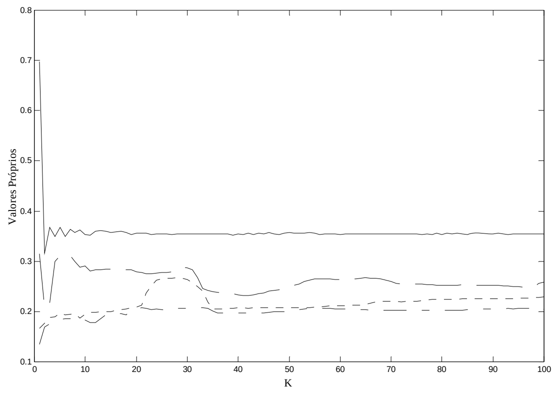

Fig. 4 – Evolução dos valores próprios ao longo do segmento.

IV. CONCLUSÕES

Do pacote de software desenvolvido destaca-se a função *GVSD* que pode ser aplicada em estudos com outros sinais (EEG, CDMA) e mesmo em imagens desde que o formato de dados seja compátivel com os parâmetros de entrada da função.

O estudo experimental realizado com sinais de áudio, revelou que a performance de algoritmo é semelhante à apresentada por outros algoritmos [6].

Pode ainda concluir-se que a evolução dos valores próprios ao longo da adaptação pode ser um indicador sobre a qualidade dos sinais recuperados.

REFERÊNCIAS

[1] Chunqi Chang, Zhi Ding et al- " A matrix pencil approach to blind source separation of colored non stationary signals"- IEEE Transactions on Signal Processing vol.48, no3 pp 900-907, 2000.

[2] James V.Stone, "Blind source separation using temporal predictability", Neural Computation, vol.13 nº7, pp 1557-1574.

[3] Ana Maria Tomé "An iterative eigendecomposition approach to blind source separation" in 3rd International Conference on Independent Component Analysis and Signal Separation, San Diego, USA, 2001.

[4] Gilbert Strang – Linear Algebra and its Applications – Saunders College Publications.

[5] Diamantaras, Kung – Principal Component Neural Networks, Theory and Applications – Wiley – 1996.

[6] Nuno Ferreira, Ana Maria Tomé – Blind Source Separation of Temporally Correlated Signals, RECPAD 2002, Aveiro, Portugal, 2002.

[7] http://www.cis.hut.fi/projects/ica/cocktail/cocktail_en.cgi

[8] http://www.ieeta.pt/~ana