SP-6
Processing audio mixtures

Włodzimierz Kasprzak

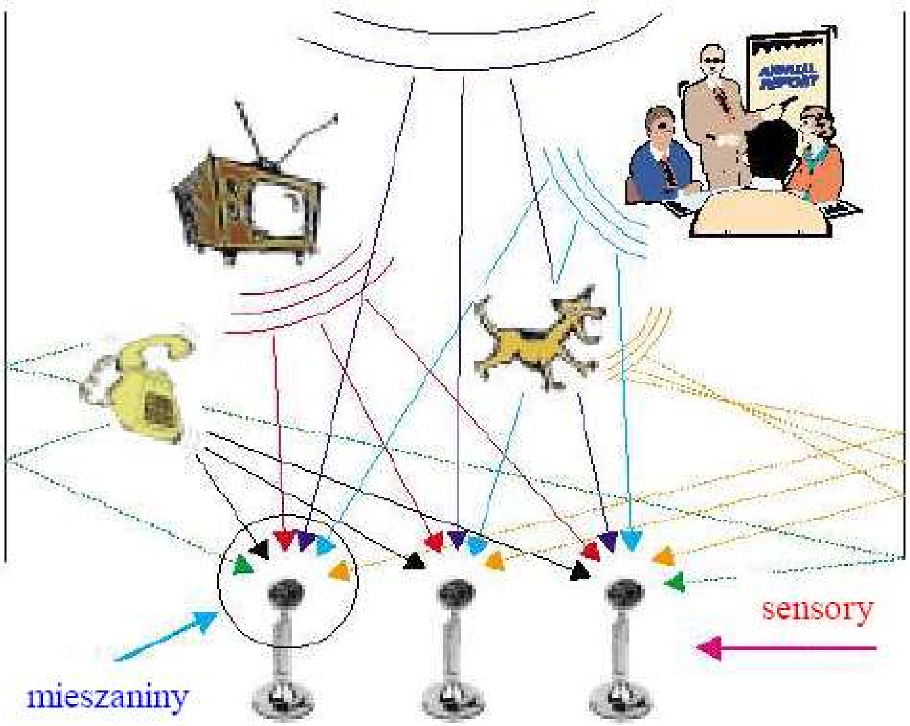
Lecture 2021

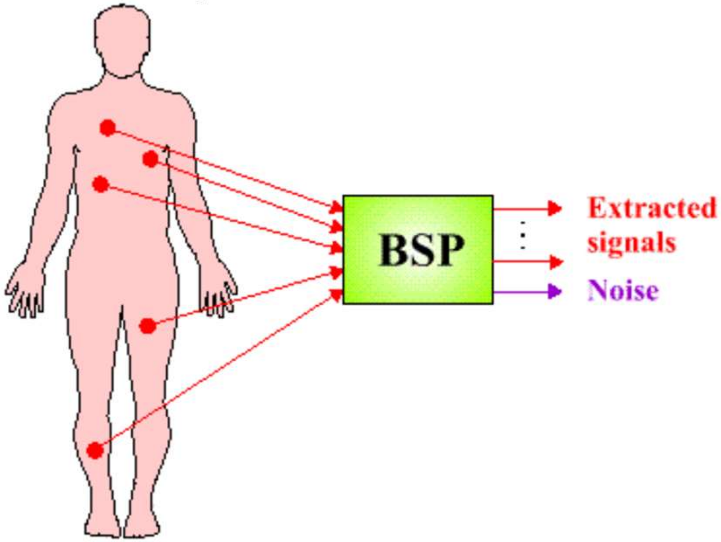
I. Blind source separation / deconvolution

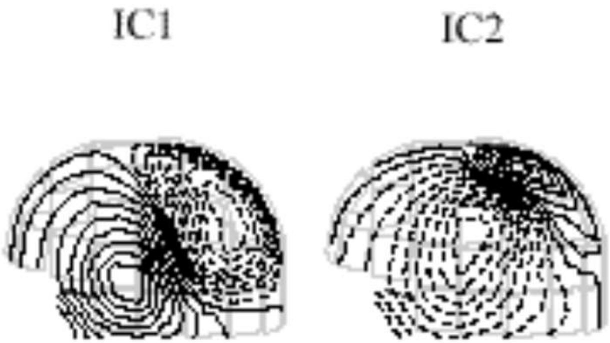
1. Signal mixtures

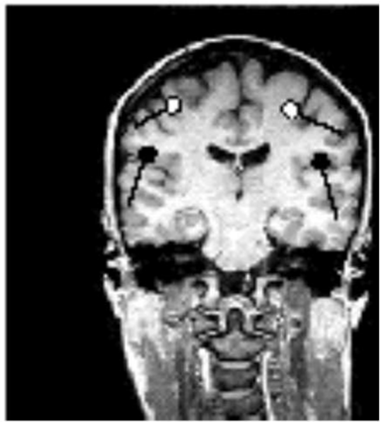
Mixtures of signals of the same type are measured: EEG, biomedical signals, audio, speech, images.

„Cocktail party” problem in audio/speech: to focus on one source (to separate it from mixtures)



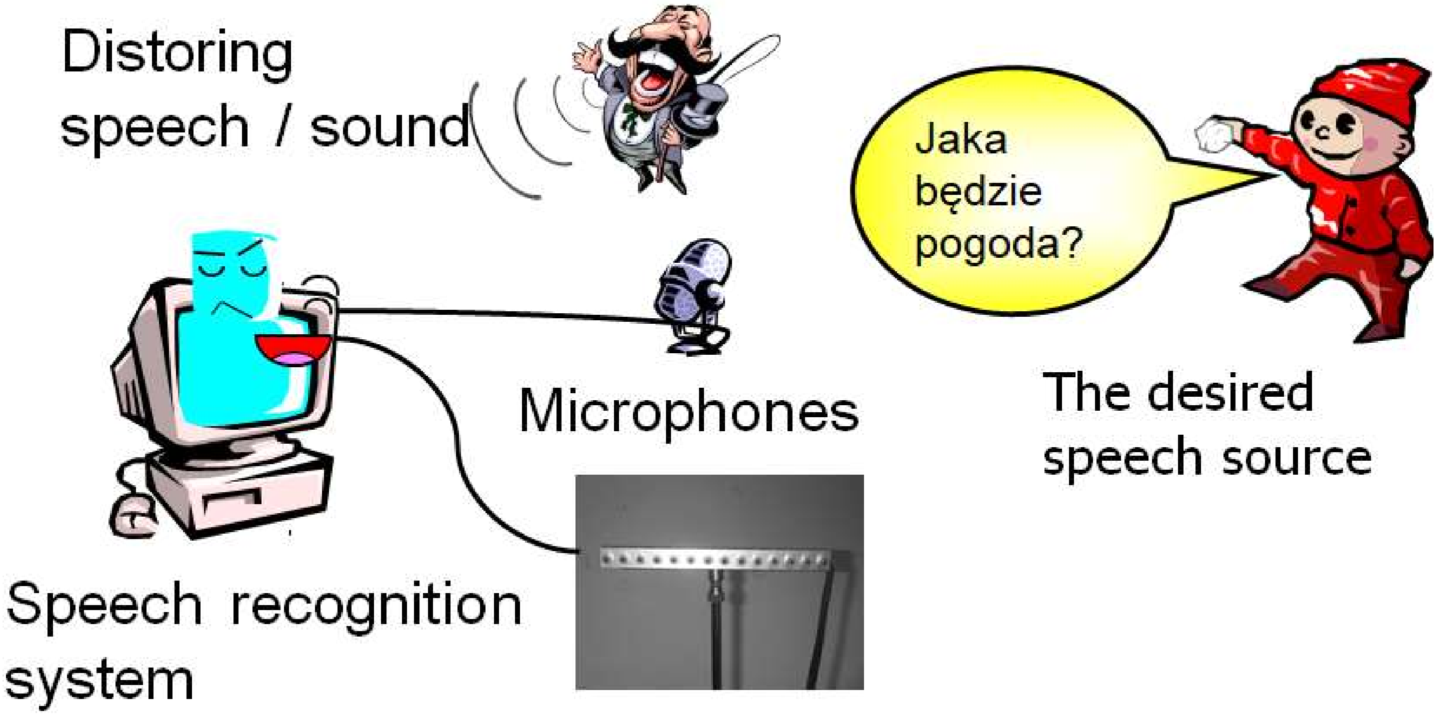






Sound mixtures

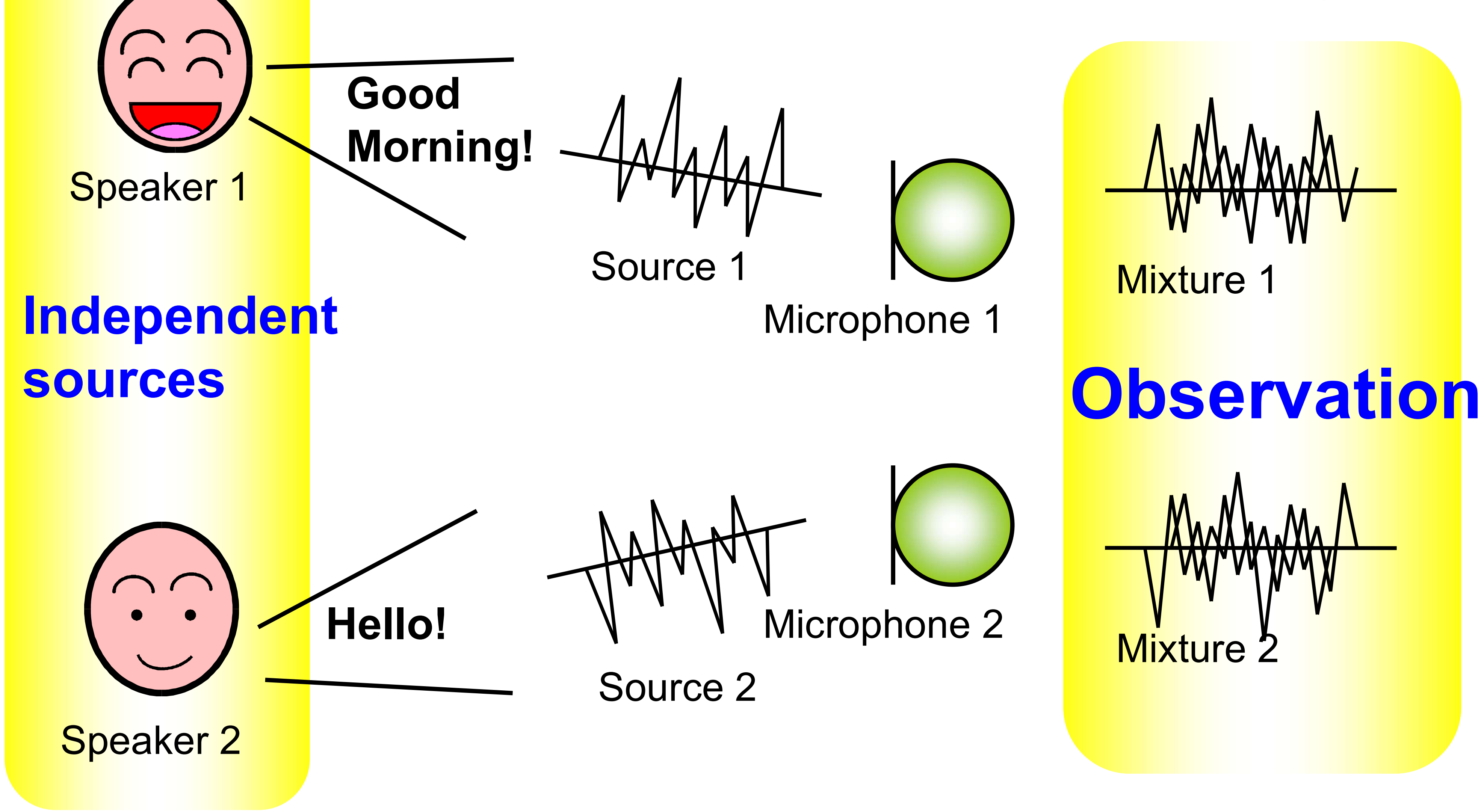
Many sensors (e.g. microphones) are employed for synchronized signal perception.



Instantaneous-time mixtures

Sources are unknown. Instantaneous-time mixtures

Solution - **unsupervised adaptive filtering**

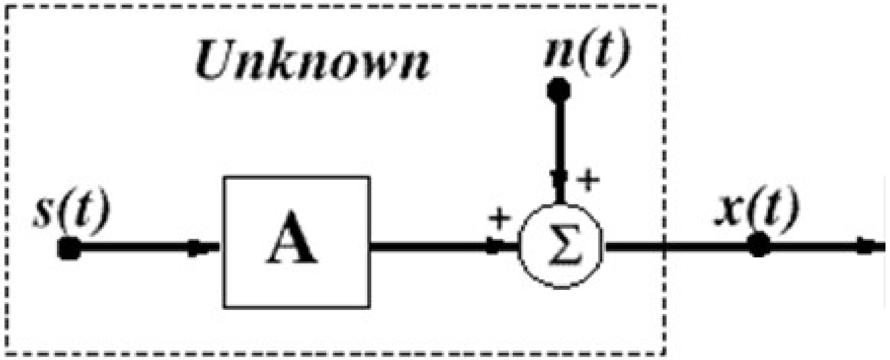


2. The inverse problem

Instantaneous-time inverse problem:

- m unknown sources (e.g. signals in time) or
  a set of *m*-dimensional data samples :
- n observed, possibly noisy, different, linear
  mixtures of the sources ($n >= m$):
- the mixing coefficients are some unknown
  constants.

$$\begin{bmatrix} s_1^T[t] \\ \vdots \\ s_m^T[t] \end{bmatrix} \quad \begin{bmatrix} x_1^T[t] \\ \vdots \\ x_n^T[t] \end{bmatrix}$$



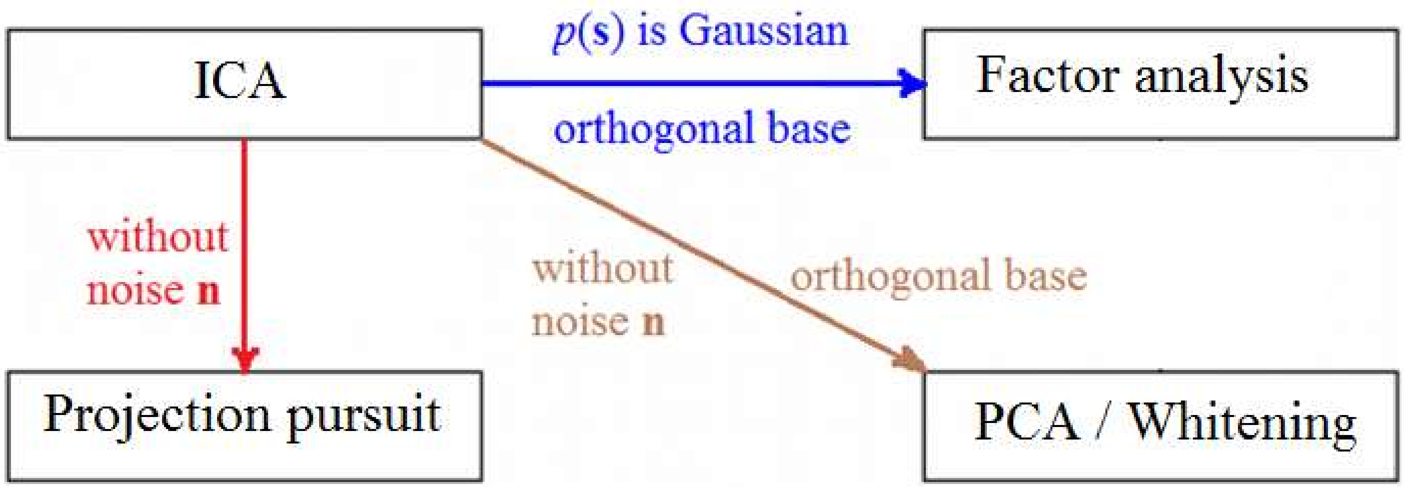
$$\mathbf{x}[t] = \mathbf{A}\,\mathbf{s}[t] + \mathbf{n}[t] = \left(\sum_{i=1}^{m} s_i[t]\, a_i \right) + \mathbf{n}[t]$$

Possible solutions

Goal: find the unknown matrix $\mathbf{A}$ and reconstruct the sources.

Possible solutions:

1. Independent component analysis → FastICA, NG BSS
2. Projection pursuit
3. Factor analysis
4. Principal component analysis → *Whitening* → AMUSE



Factor analysis, Whitening

Factor analysis (FA) estimates Gaussian distributed sources assuming that the sources (called **factors**) are mutually uncorrelated, and of unit variance: $\mathrm{E}\{\boldsymbol{s}\,\boldsymbol{s}^T\} = \boldsymbol{I}$

and noise components are assumed to be uncorrelated with each other and with the factors: $\boldsymbol{Q} = \mathrm{E}\{\boldsymbol{n}\,\boldsymbol{n}^T\}$

With above assumptions the covariance matrix of the observation is: $\mathrm{E}\{\boldsymbol{x}\boldsymbol{x}^T\} = \boldsymbol{R}_{xx} = \boldsymbol{A}\,\boldsymbol{A}^T + \boldsymbol{Q}$

Assuming $\boldsymbol{Q}$ is known or can be estimated, FA attempts to solve $\mathbf{A}$ from: $\boxed{\boldsymbol{A}\,\boldsymbol{A}^T = \boldsymbol{R}_{xx} - \boldsymbol{Q}}$

Non-stochastic model: without noise the problem simplifies to **Whitening** : $\boxed{\boldsymbol{A}\,\boldsymbol{A}^T = \boldsymbol{R}_{xx}}$

It is a specific case of PCA (Principal Component Analysis), where every „direction” in space is of unit variance.

3. AMUSE

AMUSE (Algorithm for Multiple Unknown Source Extraction) is using second-order statistics only. It consists of two steps:

- 1. Whitening
  - The data is normalized to z zero-mean set,
  - The covariance matrix without delays is computed

$$\boldsymbol{R}_{\boldsymbol{xx}}(0) = E\{\boldsymbol{x}(k)\boldsymbol{x}^{T}(k)\}$$

  - The SVD of $\boldsymbol{R}_{xx}(0)$ is computed and eigenvalues are normalized

$$\boldsymbol{C} = (\boldsymbol{R}(0)_{\boldsymbol{xx}})^{-\mathbf{1/2}} \;\; ; \;\; \boldsymbol{R}(0)_{\boldsymbol{xx}} = \boldsymbol{VLV}^{T}; \quad \boldsymbol{C} = \boldsymbol{L}^{-1/2}\boldsymbol{V}^{T}$$

- 2. Diagonalization of a time-related covariance matrix.
  - The data is whitened: $\boldsymbol{z}(k) = \boldsymbol{Cx}(k)$
  - A new covariance matrix is obtained for $\{\mathbf{z}(k)\}$ and delay lag = 1:

$$\boldsymbol{R}_{\boldsymbol{zz}}(1) = E\{z(k)z^{T}(k-1)\}$$

and is decomposed by SVD: $\boldsymbol{R}_{\boldsymbol{zz}}(1) = \boldsymbol{USV}^{T}$

AMUSE (cont.)

The final separating (demixing) matrix in AMUSE is:

$$\boldsymbol{W} = \boldsymbol{U}^{\boldsymbol{T}}\boldsymbol{C}$$

while the mixing matrix is estimated as:

$$\mathbf{A} \cong \boldsymbol{W}^{-\mathbf{1}} = \boldsymbol{C}^{\boldsymbol{T}}\boldsymbol{U}$$

The AMUSE algorithm is sensitive to noise.

This sensitivity is reduced in a modified AMUSE:

- In the second step use a linear combination of many covariance matrices obtained for different delay lags,

$$\boldsymbol{R}_{\boldsymbol{zz}} = \sum_{\boldsymbol{d}=\mathbf{1}}^{\boldsymbol{D}} \boldsymbol{a}_{\boldsymbol{d}}\boldsymbol{R}_{\boldsymbol{zz}}(d)$$

4. ICA – independent component analysis

Assumptions in ICA:

1. Sources are stochastically independent w.r.t. functions f , g that capture higher order statistics, i.e., for y_1 , y_2 it holds

$$E\{f(y_1)\,g(y_2)\} = E\{f(y_1)\} \cdot E\{g(y_2)\}$$

2. At most one of the sources is of Gaussian distribution.

Learning criteria - to maximize the non-Gaussianity or independence of sources:

1) Negentropy – difference between Entropy of Gaussian distribution (with mean and variance obtained for current data) and the entropy of current data (non-negative).

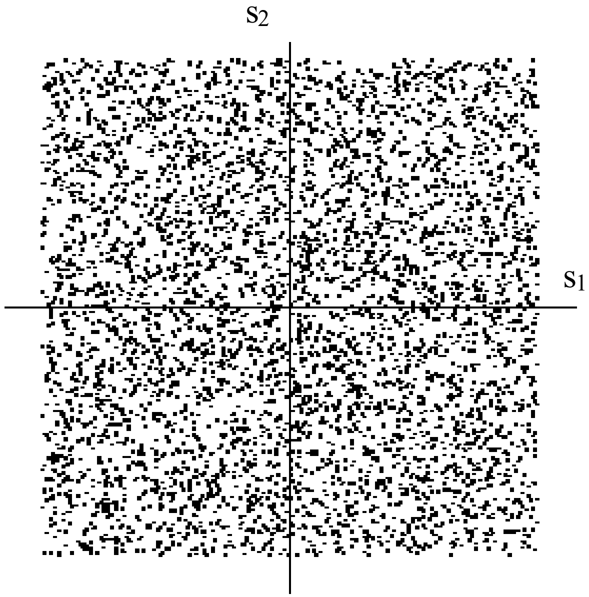
2) Kurtosis – measures the flateness of distribution – of zero value for Gaussian, positive value for a sparse distribution, negative value for a dense distribution.

ICA - example

Example [Hyvarinen et al. (2000)]

Two independent sources: $$p(s_i) = \begin{cases} \dfrac{1}{2\sqrt{3}}, & if |s_i| \leq \sqrt{3} \\ 0, & otherwise \end{cases}$$

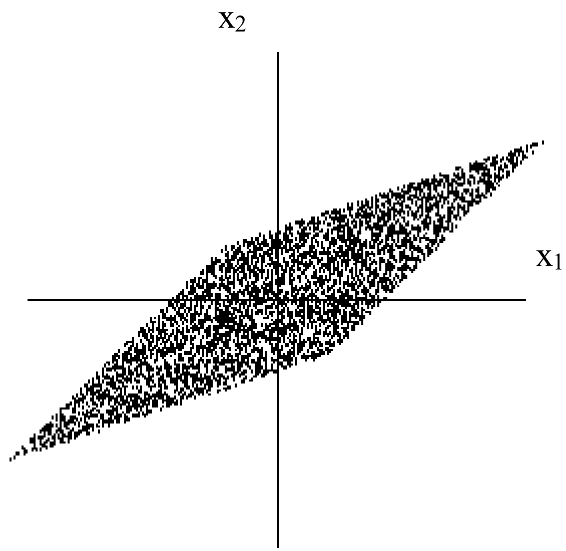
Joined distribution (s_1, s_2):



Mixing matrix

$$A_0 = \begin{bmatrix} 2 & 3 \\ 2 & 1 \end{bmatrix}$$

Mixture distribution

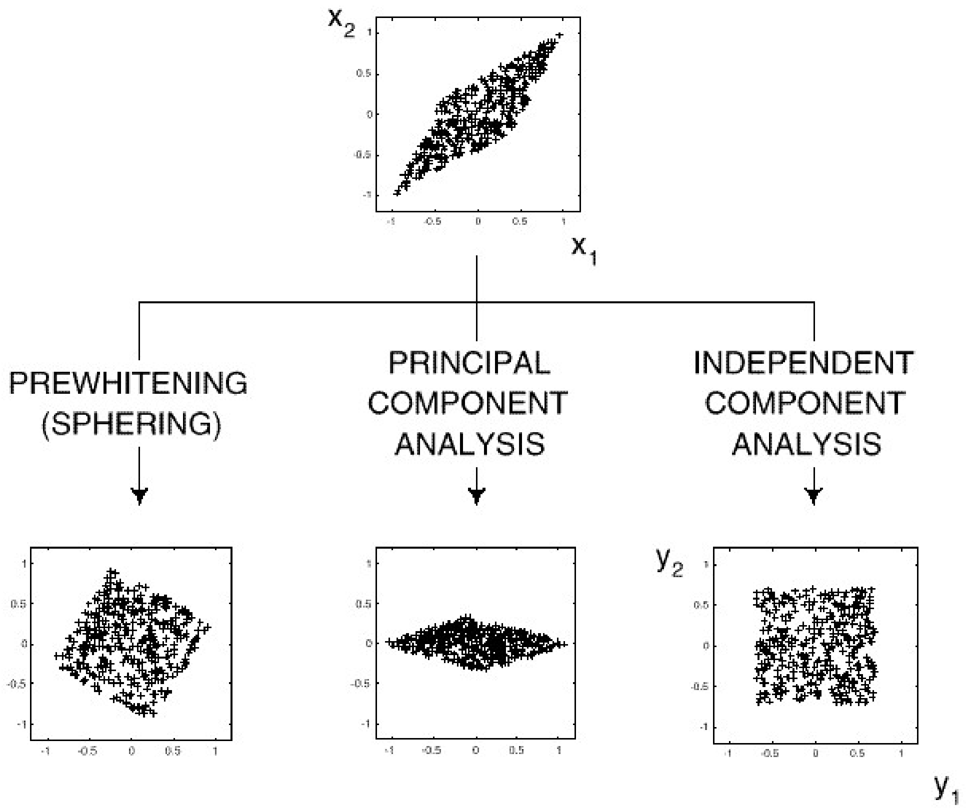


Example (cont.)

Whitening: a linear transform so that the new mixtures are uncorrelated and have a variance of one.

PCA: find orthogonal main directions → no separation.

ICA: find main directions → source separation by generalized independency.



FastICA (1)

FastICA (Hyvarinen, Karhunen, Oja) is a batch method for ICA. It requires that the mixtures are **centered** and **whitened** first.

Centering: $$\boldsymbol{x}_{centered} = \boldsymbol{x} - E\{\boldsymbol{x}\} = \boldsymbol{x} - \boldsymbol{m}_x$$

Whitening: $$E\{\mathbf{x}\mathbf{x}^T\} = \mathbf{R}_{xx} = \mathbf{U}\Lambda\mathbf{U}^T$$

$$\mathbf{x}_{whitened} = \Lambda^{-1/2}\mathbf{U}^T\mathbf{x}$$

Then the method iteratively updates the weight matrix $\mathbf{W}$ – vector-by-vector, while maximizing the non-Gaussianity of the projection $\mathbf{W}^T\mathbf{x}$ (see next page).

FastICA (2)

A. Initialize a nonzero matrix: $\mathbf{W}=[\mathbf{w}_1,\mathbf{w}_2,\ldots,\mathbf{w}_n]$

B. Iterate

1) FOR outputs $p=1,2,\ldots,n$ DO

2) Weight update: $\boldsymbol{w}_p^+ = E\{\boldsymbol{x}\, g(\boldsymbol{w}_p^T\, \boldsymbol{x})\} - E\{(g'(\boldsymbol{w}_p^T\, \boldsymbol{x})\}\boldsymbol{w}_p$

where g is a nonlinear function, g' – its first derivation.

3) Normalize to unitary length: $\boldsymbol{w}_p = \dfrac{\boldsymbol{w}_p^+}{\left\|\boldsymbol{w}_p^+\right\|}$

4) A Gram-Schmidt orthogonalization and normalization:

$$\mathbf{w}_p' = \mathbf{w}_p - \sum_{j=1}^{p-1} \mathbf{w}_p^T \mathbf{w}_j \mathbf{w}_j^T \qquad\qquad \mathbf{w}_p = \frac{\mathbf{w}_p'}{\left\|\mathbf{w}_p'\right\|}$$

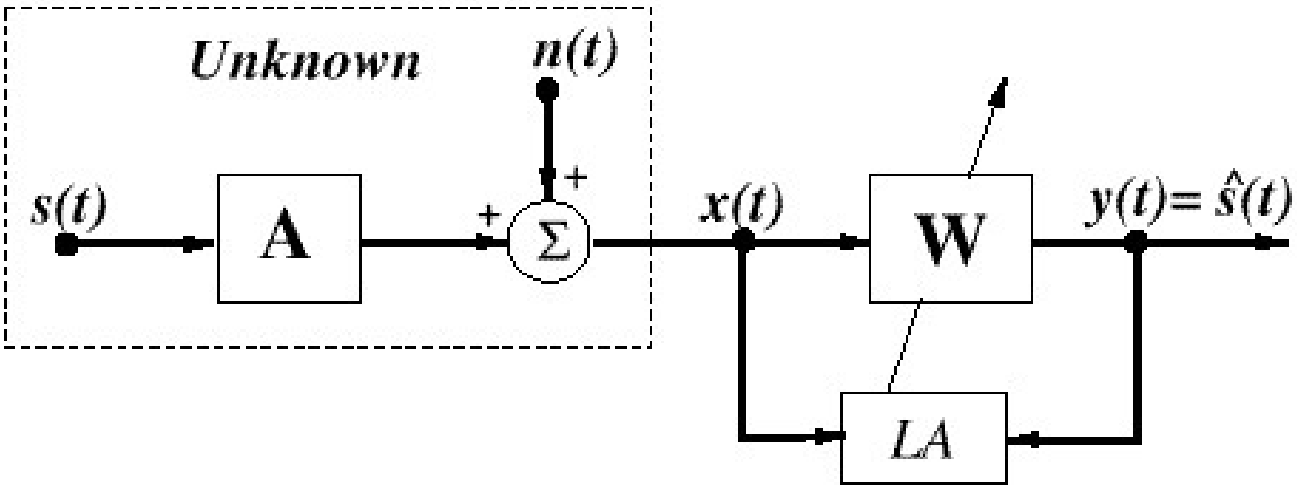
5) IF $\mathbf{W}$ has not yet converged THEN next iteration of 1)-5).

Adaptive ICA (BSS)

Blind source separation (BSS): an $m \times n$ separating matrix $\mathbf{W}(t)$ is updated so that the m-vector, $\boldsymbol{y}(t) = \mathbf{W}(t)\,\boldsymbol{x}(t)$, becomes an estimate, $\mathbf{y}(t) \approx \mathbf{s}(t)$, of the original independent sources up to **scaling** and **permutation indeterminacy**.

If source scaling ($\mathbf{S}$) and permutation ($\mathbf{P}$) are known, then:

$$\mathbf{W\,A = S\,P\,I}$$



KLD criterion

The **Kullback–Leibler divergence** measures the distance between two distributions. It can be applied as a criterion for for BSS to measure the dependency among output signals (mutual information):

$$D(\boldsymbol{y} \,||\, \{y_k\};\boldsymbol{W}) = \int p(\boldsymbol{y}) \log \frac{p(\boldsymbol{y})}{\prod_{k=1}^{K} p(y_k)} d\boldsymbol{y}$$

$$D(\boldsymbol{y} \,||\, \{y_k\},\boldsymbol{W}) = -H(\boldsymbol{y};\boldsymbol{W}) + \sum_{k=1}^{K} H(y_k)\,; \quad D(\boldsymbol{y} \,||\, \{y_k\};\boldsymbol{W}) \geq 0$$

$H(\boldsymbol{y};\, \mathbf{W})$ - the average information of the joint output $\mathbf{y}$.

The $H(y_k)$-s are entropies of marginal distributions – their sum is constant – only the information related to joint output distribution, $H(\mathbf{y},\, \mathbf{W})$, is changed during optimization.

The BSS optimization rule: $\arg\min_{\boldsymbol{W}} D(\boldsymbol{y} \,||\, y_k\};\boldsymbol{W})$

Stochastic gradient descent BSS

Update rule - move along the negative gradient of the goal function:

$$\boldsymbol{W}(t+1) = \boldsymbol{W}(t) - \eta(t)\frac{\partial D}{\partial \boldsymbol{W}}$$

$$-\frac{\partial D(\boldsymbol{W})}{\partial \boldsymbol{W}} \propto \left((\boldsymbol{W}^{\mathrm{T}})^{-1} - \int p(x)\phi(\boldsymbol{y})\boldsymbol{x}^{\mathrm{T}} d\boldsymbol{x}\right)$$

$$= \left((\boldsymbol{W}^{\mathrm{T}})^{-1} - \mathrm{E}_x\left[\phi(\boldsymbol{y})\boldsymbol{x}^{\mathrm{T}}\right]\right)$$

$$= \left(\boldsymbol{I} - \mathrm{E}_y\left[\phi(\boldsymbol{y})\boldsymbol{y}^{\mathrm{T}}\right]\right)(\boldsymbol{W}^{\mathrm{T}})^{-1}$$

Where

$$\phi(\boldsymbol{y}) \equiv \left[\frac{\partial \log p(y_1)}{\partial y_1}, \ldots, \frac{\partial \log p(y_K)}{\partial y_K}\right]^{\mathrm{T}}$$

„Natural gradient” BSS

The **adaptive separation rule,** generated according to the **natural gradient**:

$$\boldsymbol{W}(t+1) = \boldsymbol{W}(t) - \eta(t)\frac{\partial D}{\partial \boldsymbol{W}}\boldsymbol{W}^T\boldsymbol{W}$$

is:

$$\boxed{\boldsymbol{W}(t+1) = \boldsymbol{W}(t) + \eta(t)\{\boldsymbol{I} - f[\boldsymbol{y}(t)]\cdot g[\boldsymbol{y}^T(t)]\}\cdot\boldsymbol{W}(t)}$$

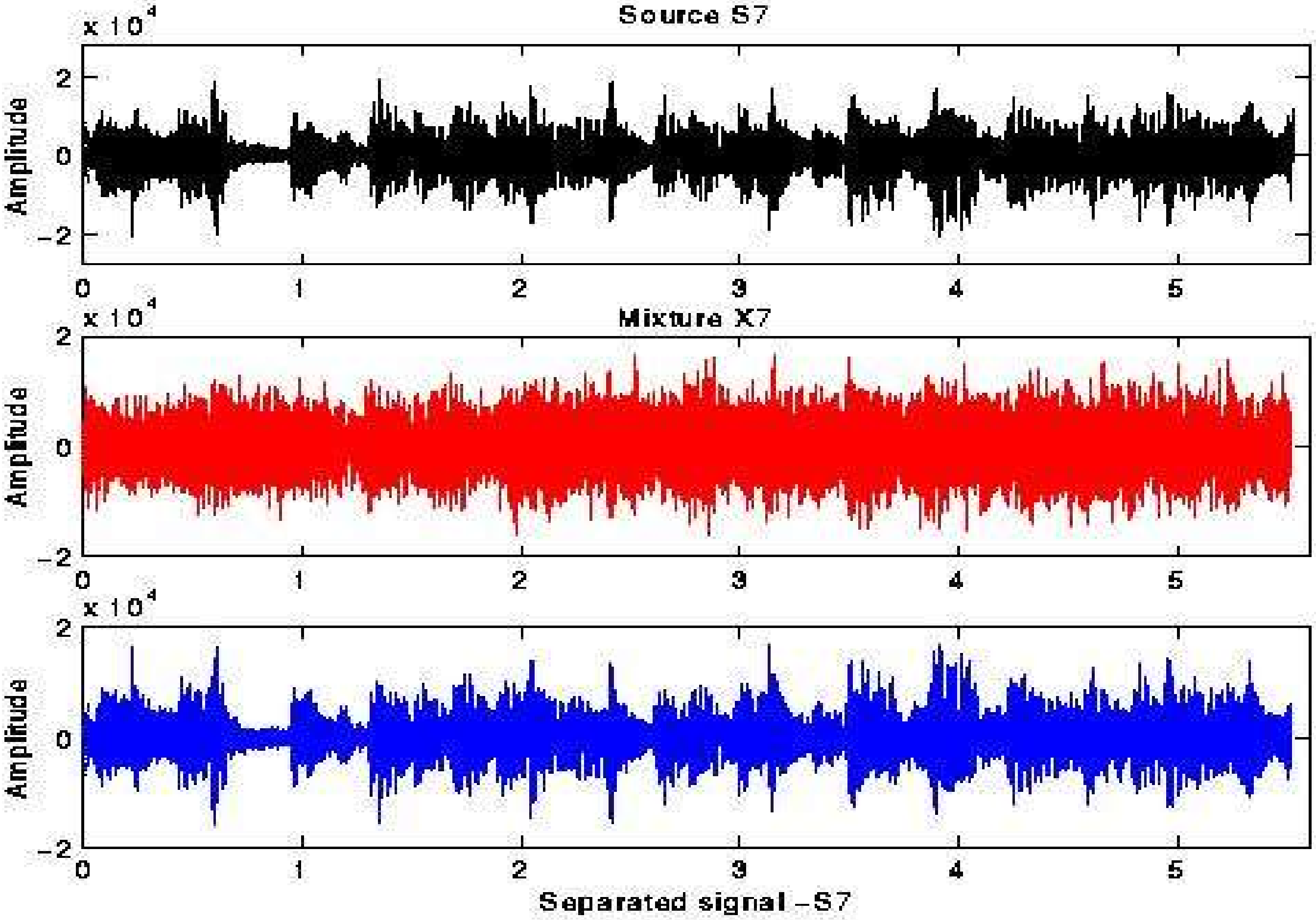
$f(\mathbf{y}) = [f(y_1), \ldots, f(y_n)]^T$ and $g(\mathbf{y}^T) = [g(y_1), \ldots, g(y_n)]$ are vectors of non-linear activation functions, which approximate **higher–order** moments of the signals**.**

If the source has a negative normalized **kurtosis** value (i.e. it is a *sub–Gaussian* signal), we choose: $f(y) = y^3,\ g(y_j) = y_j$.

For a *super–Gaussian* source with positive kurtosis, we choose: $f(y) = \tanh(\alpha\, y),\ g(y_j) = y_j$.

Example

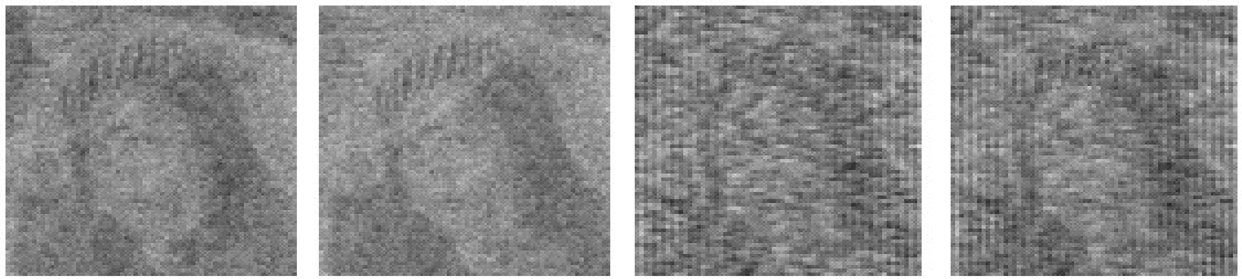
Sound separation: one unknown source, one mixture and one separated signal are shown:



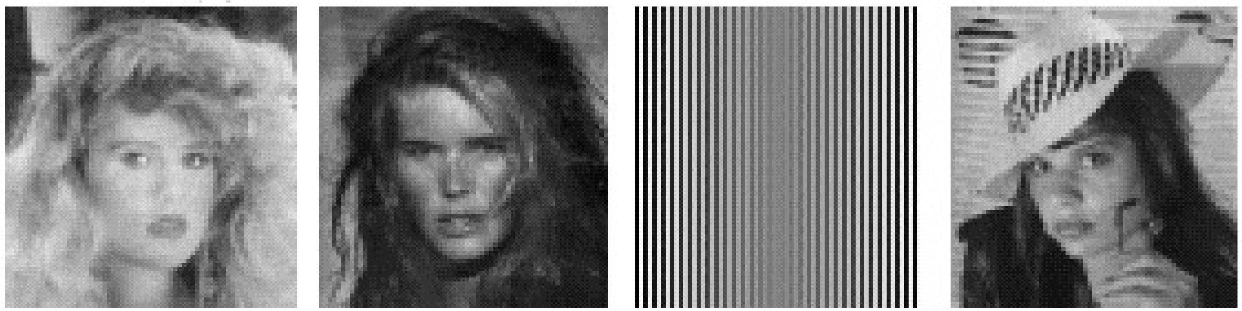
Example

Four unknown sources:

Four mixtures with added noise:



Separated signals:



5. The separation quality

(A) If the source signals ***are known***

For every pair of zero-mean signals, (*output* Y_i , *source* S_j), of amplitudes scaled to $<-1, 1>$, compute its SNR[i,j] (signal to noise ratio) as:

$$\mathrm{SNR}[i, j] = \langle S_i^2 \rangle / \mathrm{MSE}[i, j],$$

where $\langle S_i^2 \rangle$ - the time-average of 2-nd power of source *i* (i.e. the average energy), $\mathrm{MSE}[i, j]$ - the mean square error of approximating S_j by $\pm Y_i$, $i=1,\ldots,n;\ \ j=1,\ldots,m$:

$$MSE[i, j] = \frac{1}{N} \sum_{k=0}^{N-1} [S_j(k) - Y_i(k)]^2 = E\{(S_j - Y_i)^2\}$$

where N is the number of samples (and the smaller error of comparing S_j with $\{+Y_i , -Y_i \}$ is taken)

A matrix $\mathbf{P}$=[$p_{i,j}$] is created, with $p_{i,j} = SNR[i, j]$.

The error index:
$$\boxed{EI(\mathbf{P}) = \frac{1}{m}\left(\sum_i \sum_j |\,\tilde{p}_{i,j}\,| - n\right) + \frac{1}{n}\left(\sum_j \sum_i |\,\bar{p}_{i,j}\,| - m\right)}$$

Measuring the quality

Every row i of $\mathbf{P}$ is scaled: $\widetilde{\mathbf{P}} = Norm(\mathbf{P})$, such that $\forall i\ (\max_j(\widetilde{a}_{i,j}) = 1)$

Every column j is scaled: $\overline{\mathbf{P}} = NormCol(\mathbf{P})$, such that $\forall j\ (\max_i(\overline{a}_{i,j}) = 1)$

(B) If the mixing matrix $\mathbf{A}$ is **known**

The error index for $EI(\mathbf{P})$ is defined like before, but now the matrix $\mathbf{P}$ is obtained as: $\mathbf{P} = \mathbf{W}\ \mathbf{A}$.

The entries $p_{i,j}$-s of matrix $\mathbf{P}$ are again normalized along rows (i=1, ..., n) or columns (j=1,...,m).

Ideal case of perfect separation:

- $\boldsymbol{P}$ becomes a permutation matrix.
- Only one element in each row and column equals to unity, and all the other elements are equal to zero.
- Then the minimum of EI is 0.

Measuring the quality (cont.)

(C) If both the sources and mixing matrix are unknown

The normalised **mutual correlation** coefficients are computed

$$r_{i,j} = \frac{f(y_i)\cdot g(y_j)}{|f(y_i)||g(y_j)|}$$

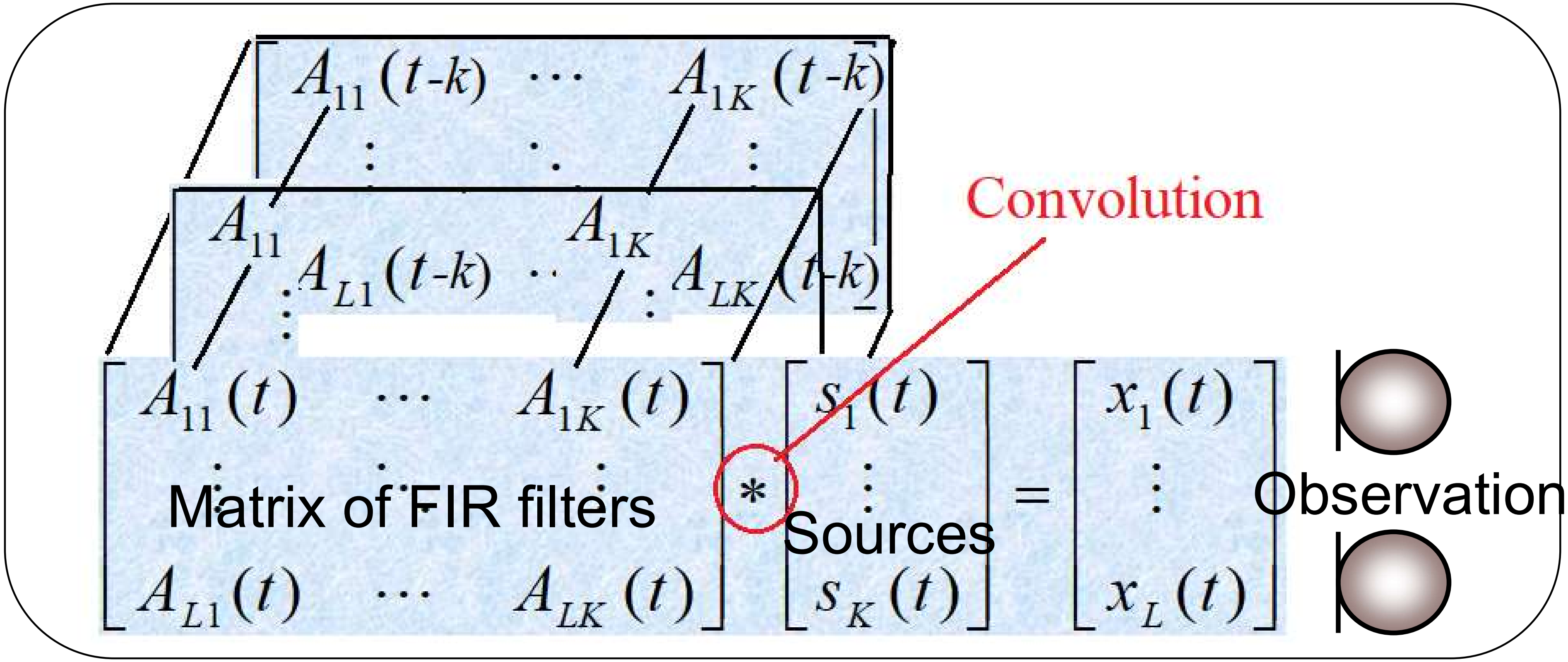
for every pair of output signals y_i and y_j , giving the square matrix: $\mathbf{P}=[r_{i,j}]$.

The error index for the set of separated sources is computed as:

$$\boxed{EI(\widetilde{\boldsymbol{P}}) = \frac{1}{n}\left(\sum_i \sum_j |r_{i,j}| - n\right)}$$

6. MBD

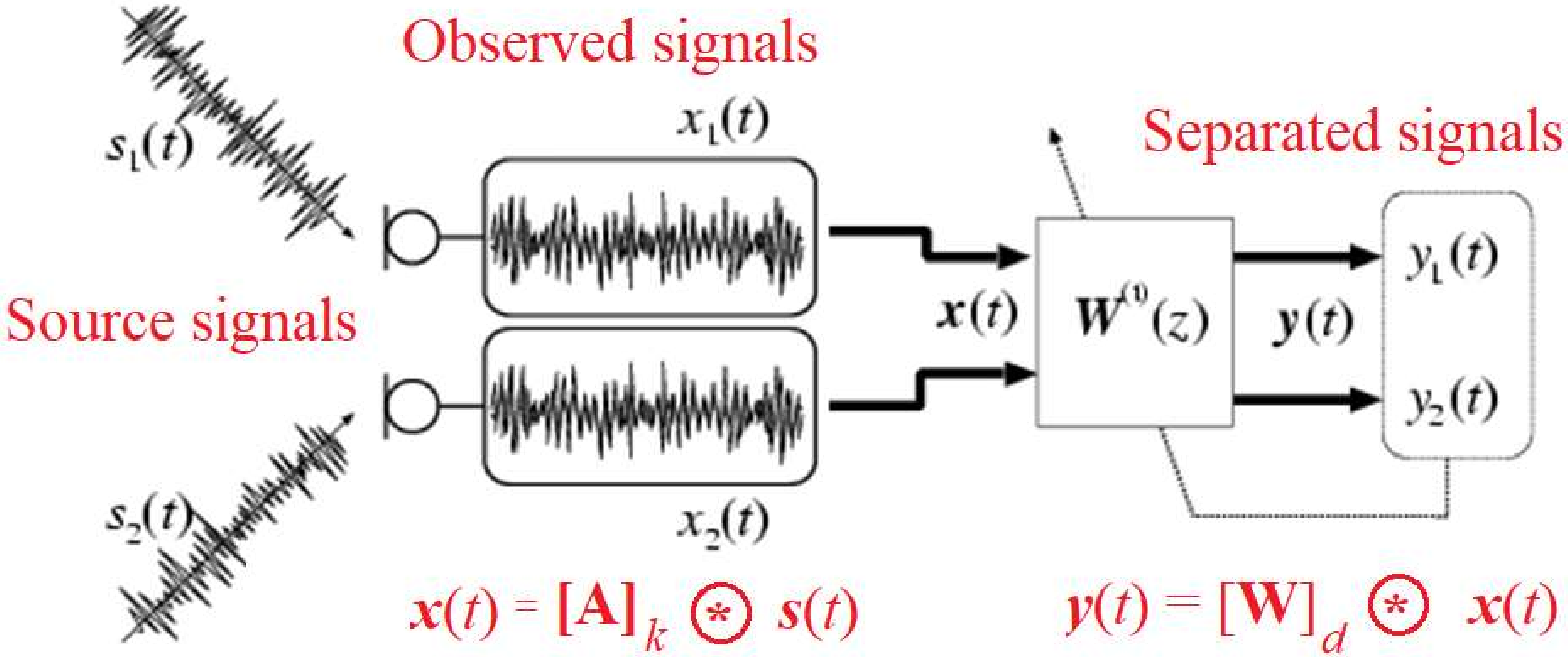
The convolutive mixtures:



Goal of **MBD** (multichannel blind signal deconvolution):

- estimate a multi-input set of FIR filters - the inverse to unknown filters performing convolutive mixtures.

Time-domain MBD

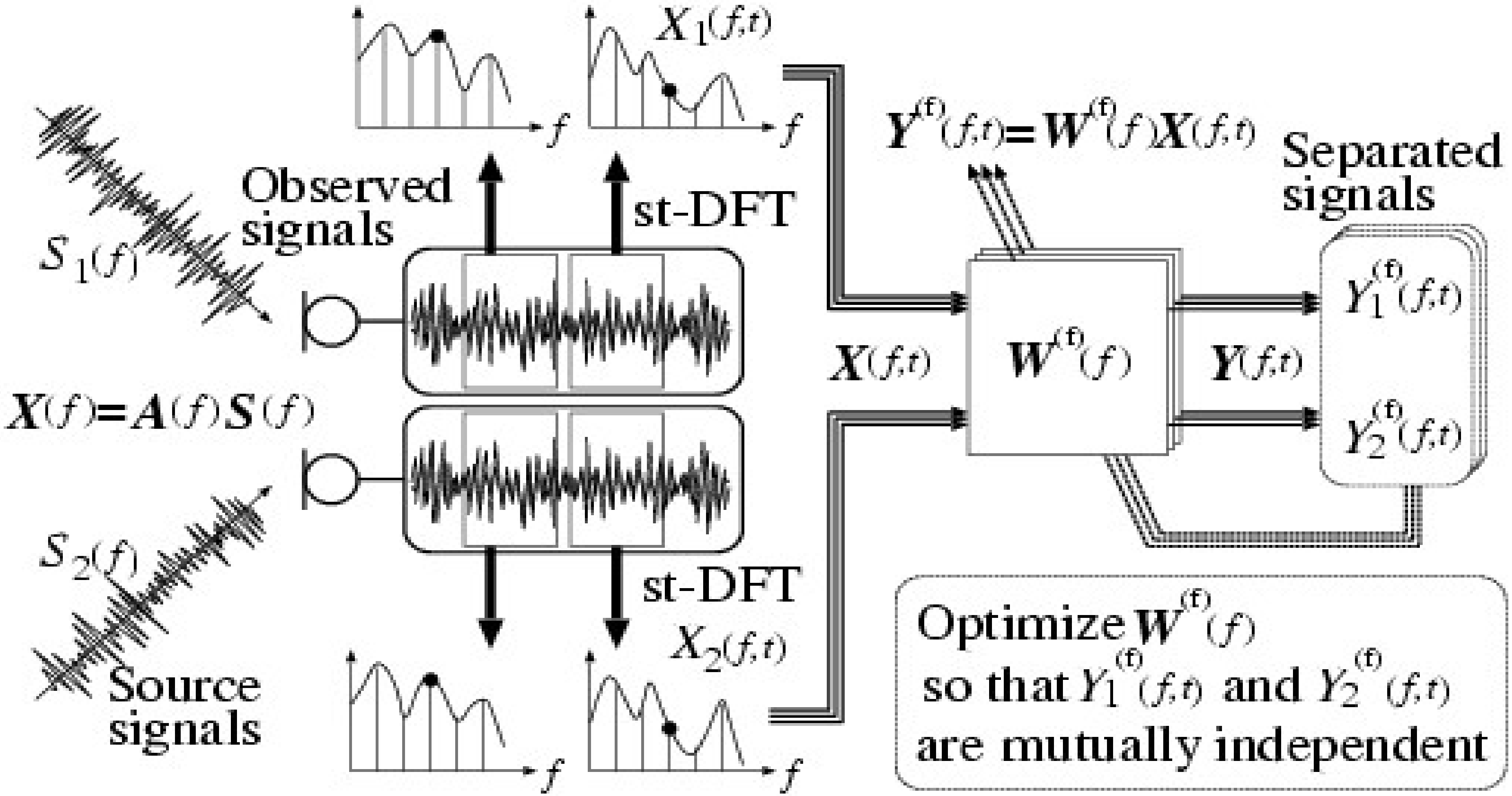


Problems:

- high complexity (~ 2400 delay coefficients per FIR),
- non-zero auto-correlations of sources across time.

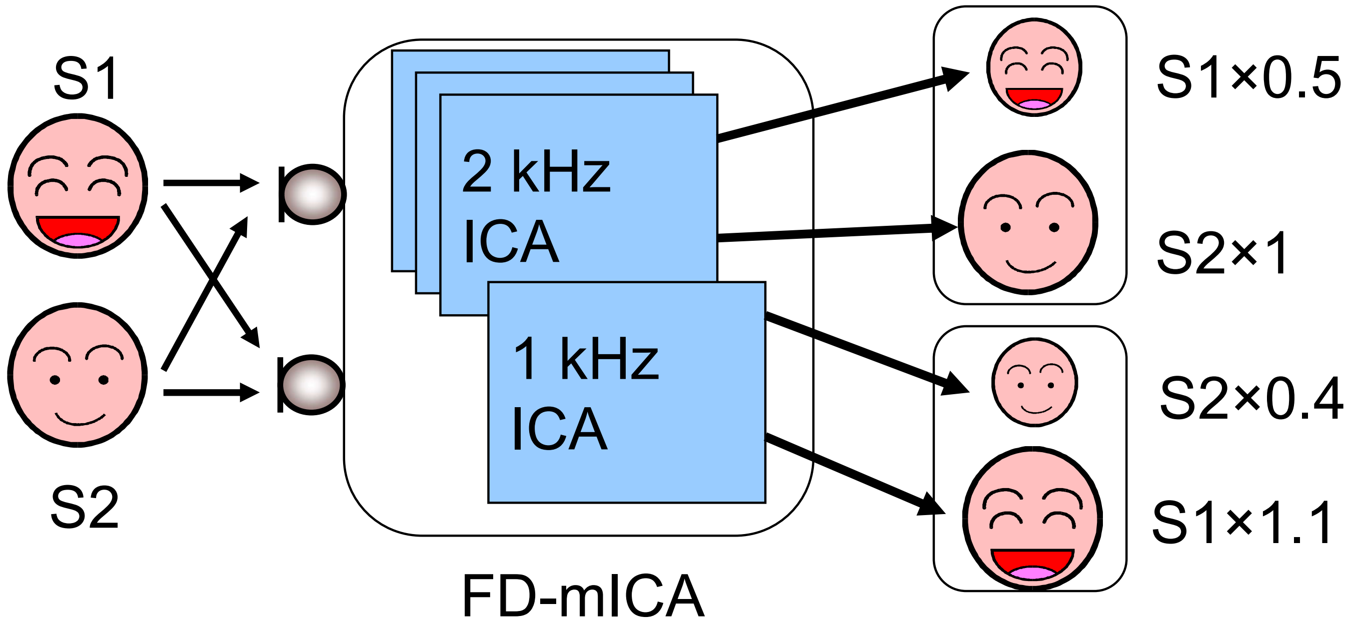
Frequency-domain MBD

Frequency-Domain MBD - multiple ICA separations (one demixing per frequency bin)



Problems of multi-ICA

Problems: unknown ICA output permutations and amplitude scales for each frequency bin.



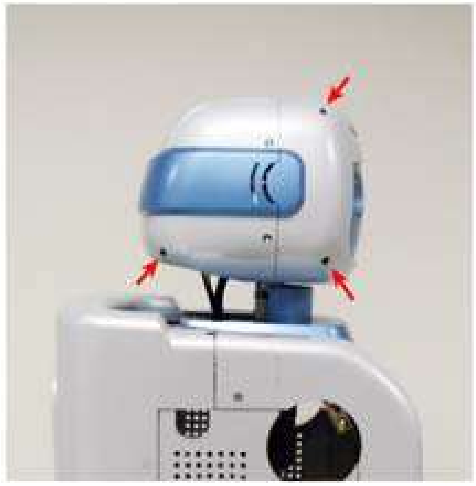
Approaches for permutation detection:

- testing the output correlations (Murata et al.);
- exploring a “direction pattern“ *W* (Kurita & Saruwatari);
- correlate the separation matrices for neighbour bins (Parra et al. , Asano et al.)

II. Auditory scene analysis

Application scenario

1. **Human speech source localization in the neighborhood of a mobile robot**

@ U-H Kim et al. KIST

2. **Audio control in a room**

1. Source detection and localization

Source **detection** / localization principle:

- Time difference of arrival: phase-difference depends on direction of source,
- Attenuation ratio: depends on distance of source.

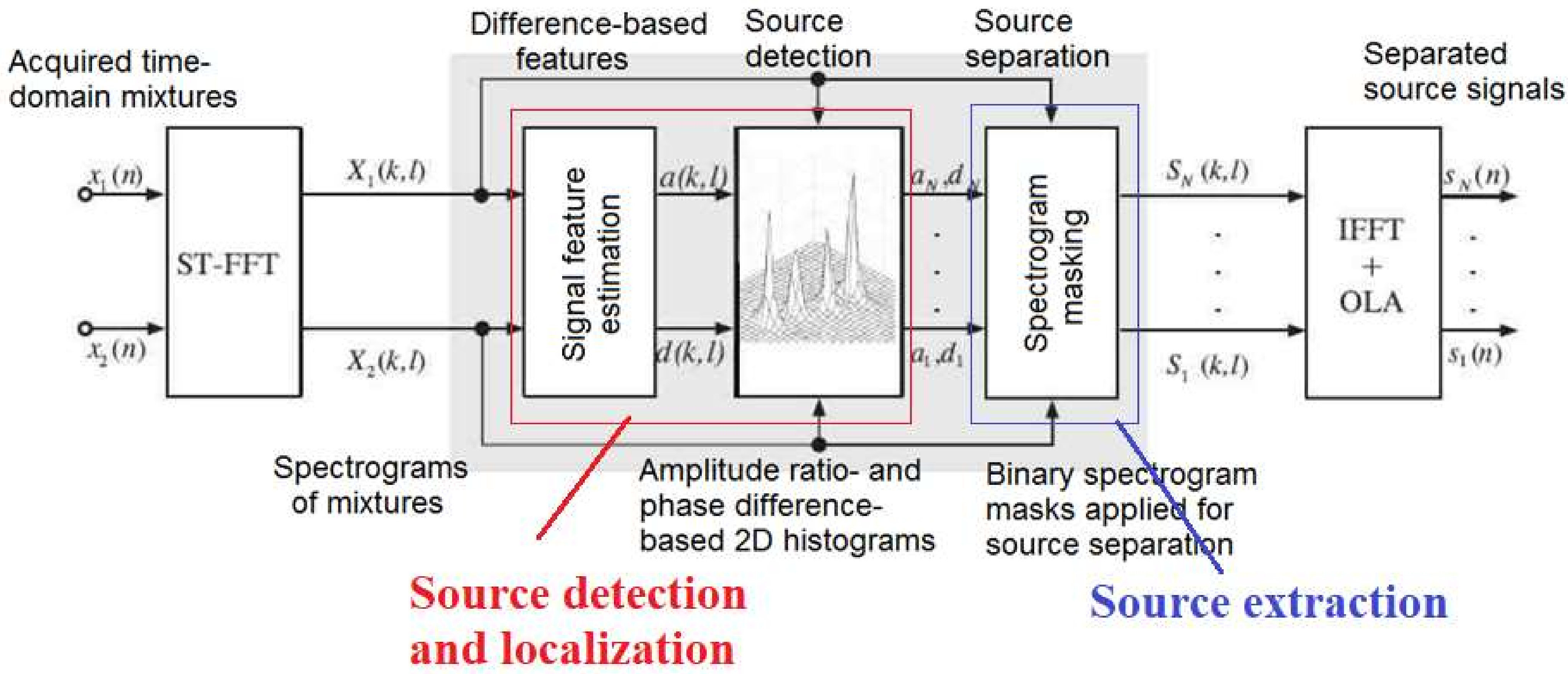
Source **extraction** principle:

WDO – disjoint orthgonal sources in the frequency domain → spectrogram masking (one spectrogram mask per source)

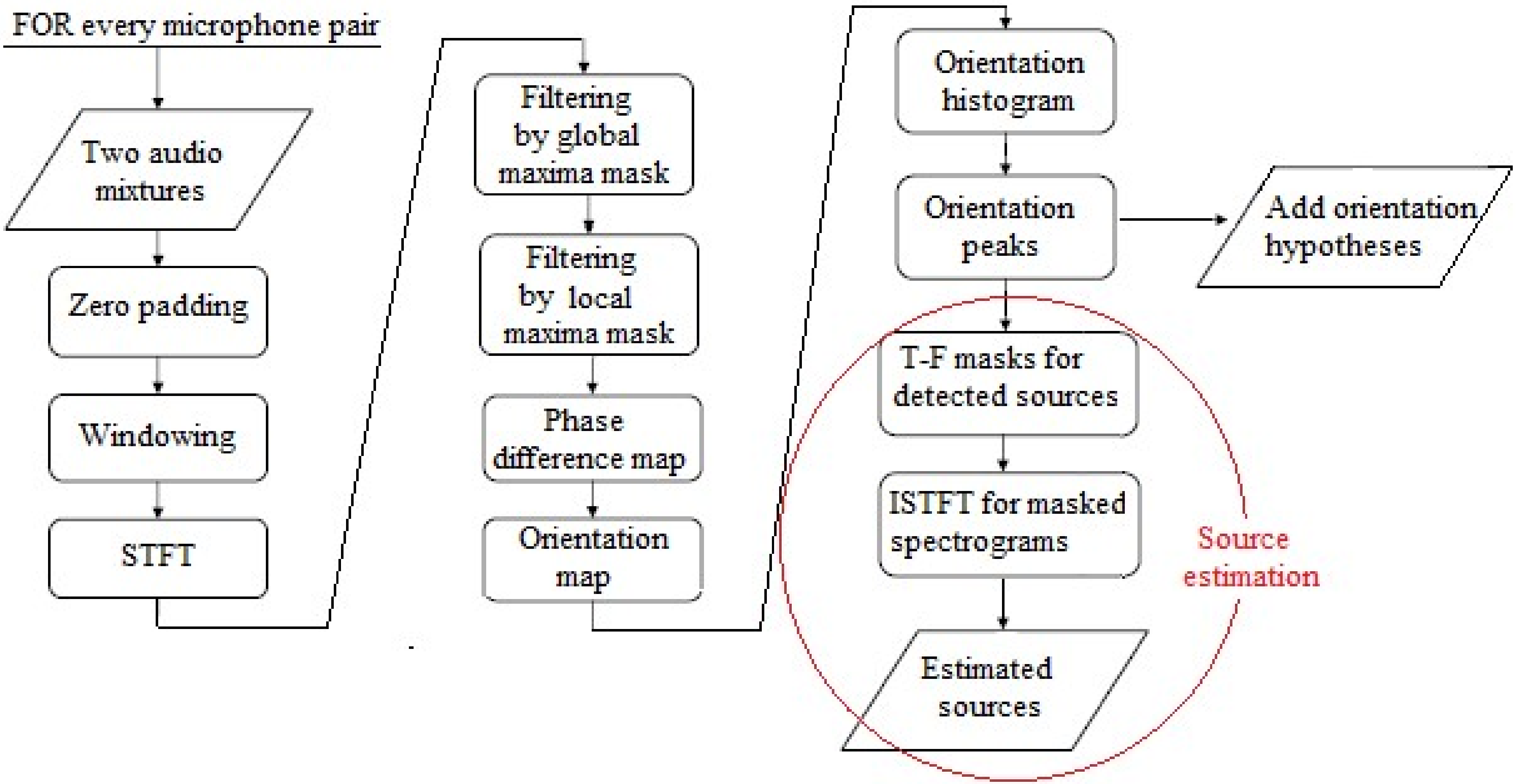
Solution scheme

A) Source **detection / localization**

B) Source **extraction**

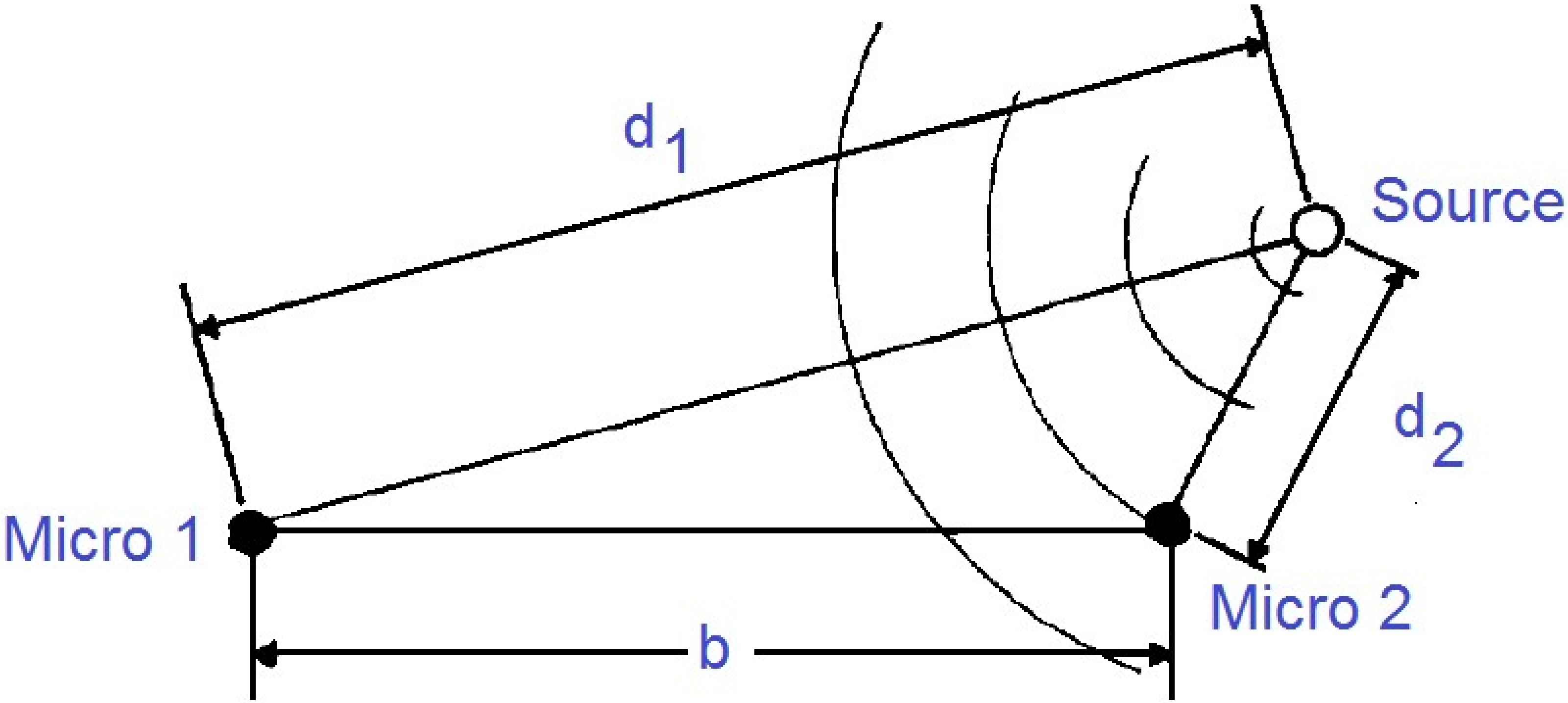


Source detection and extraction



2. Basics – TDA, WDO

- Basic element – pair of microphones



- Mixing of 2 sources in the frequency (DFT) domain

$$\begin{bmatrix} X_1(t,f) \\ X_2(t,f) \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ e^{-j\frac{2\pi f \delta_1}{L}} & e^{-j\frac{2\pi f \delta_2}{L}} \end{bmatrix} \begin{bmatrix} S_1(t,f) \\ S_2(t,f) \end{bmatrix}$$

WDO

- Basic assumption (WDO):

Disjoint orthogonal signals in the frequency domain:

$$S_i(t,f)\cdot S_j(t,f)\approx 0 \quad , \quad \forall i\neq j, \forall(t,f)$$

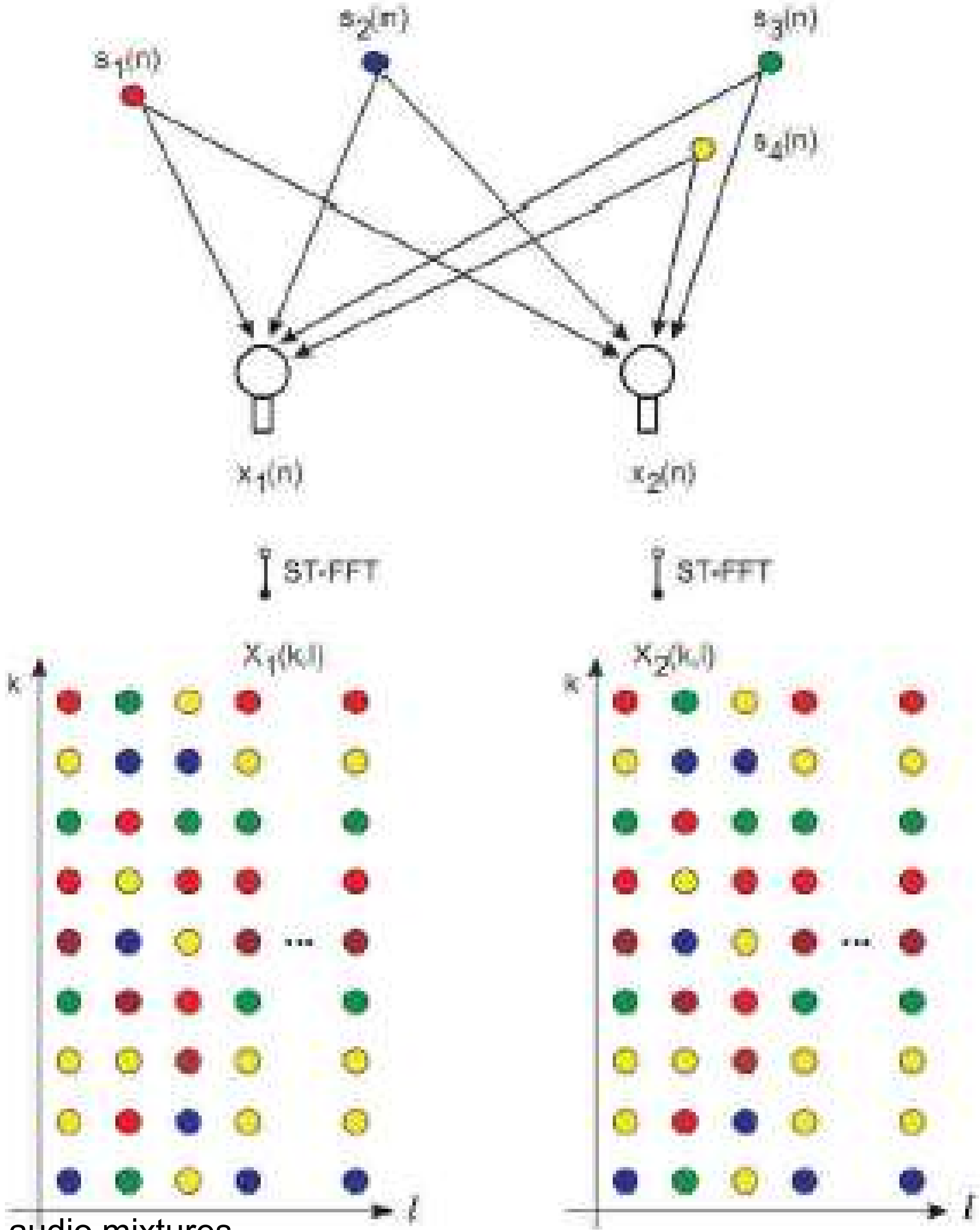
- The contribution of each particular source to the mixtures can be separated in the frequency domain. For two sources:

$$\begin{bmatrix} X_1(t,f) \\ X_2(t,f) \end{bmatrix} = \begin{bmatrix} 1 \\ e^{-j\frac{2\pi f\delta_i}{L}} \end{bmatrix} S_i(t,f)$$

WDO (2)

Illustration of WDO :

4 sources, 2 mixtures



→

Every mixture can be decomposed in the same way into 4 mutually orthogonal spectrograms

Time delay –> phase shift

The delay δ_i is related to a phase function:

$$\delta(t,f) = \frac{L}{2\pi f}\phi(t,f)$$

where $\phi(t,f)$ is the phase difference

$$\phi(t,f) = \angle X_1(t,f) - \angle X_2(t,f)$$

Source detection (DUET method)

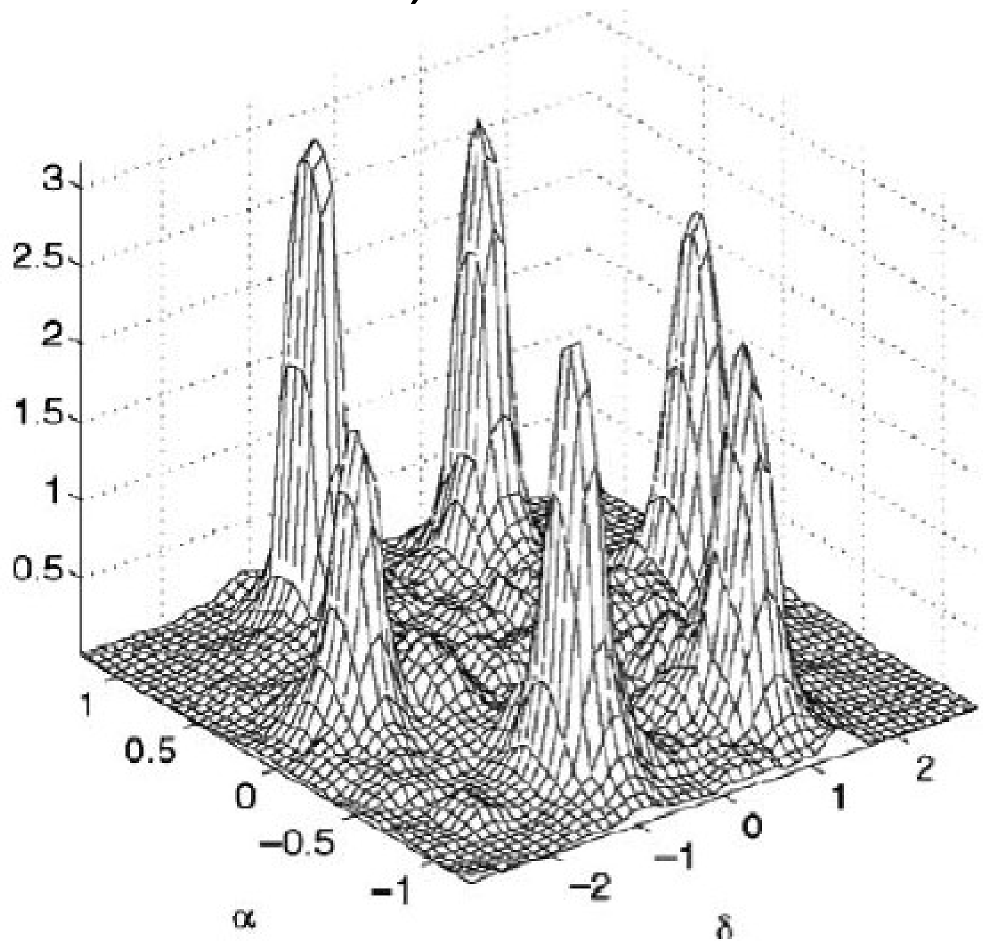
A 2D histogram (time-delay, attenuation ratio).

Clustering of points.

Detecting numer of clusters

Computing center of masses

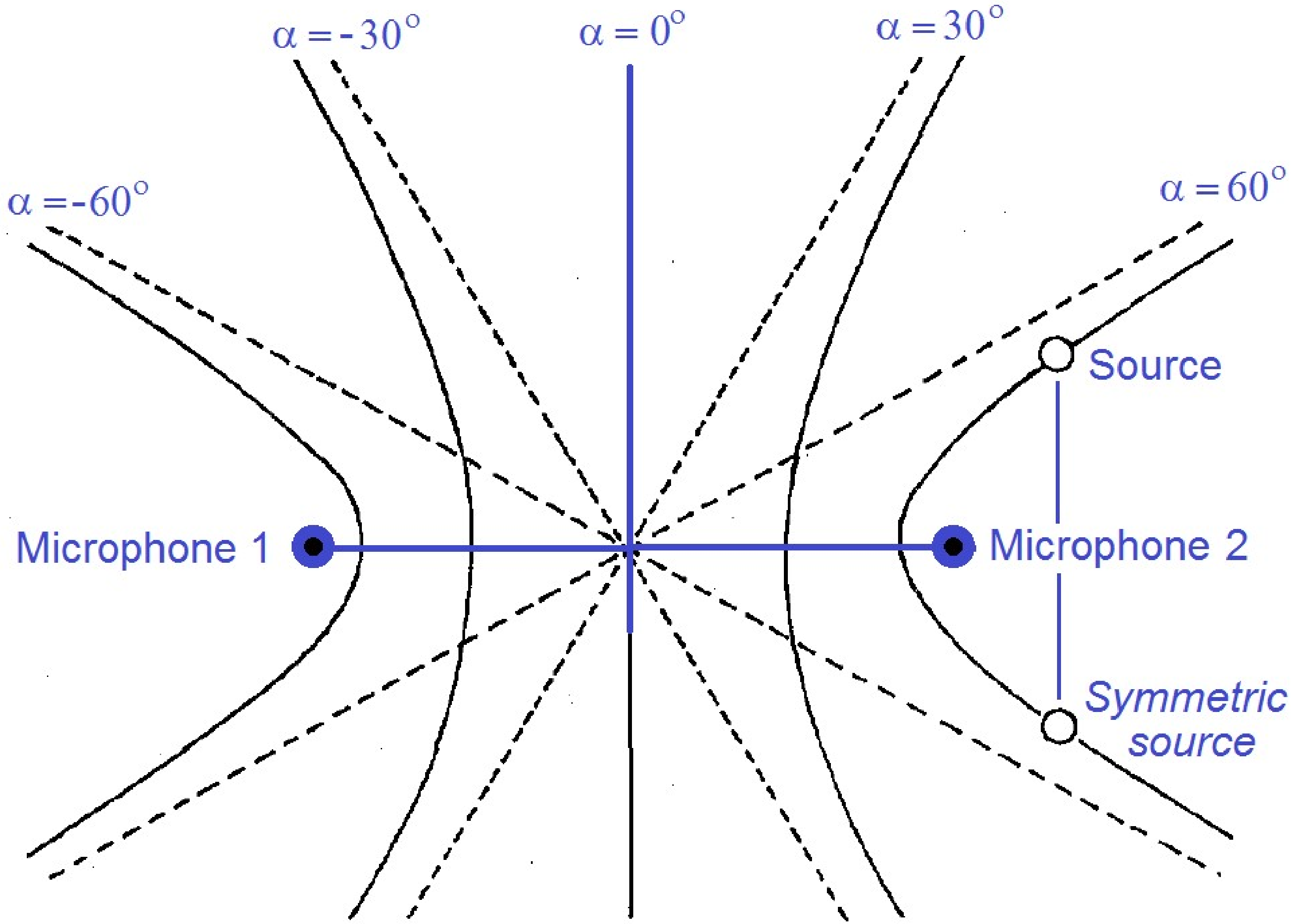
[DUET by Yilmaz & Rickard, 2004]



If the attenuation ratio provides no additional discriminate information → 1D histogram.

Direction ambiguity

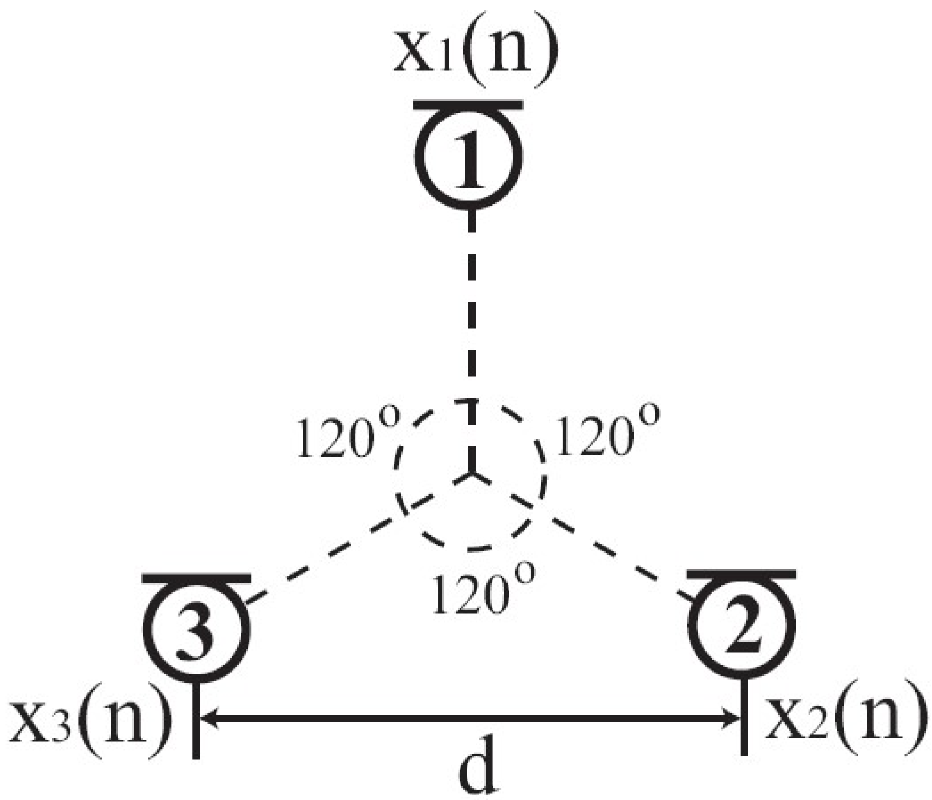
With two microphones „symmetric” directions appear:



3. Triangle of microphones

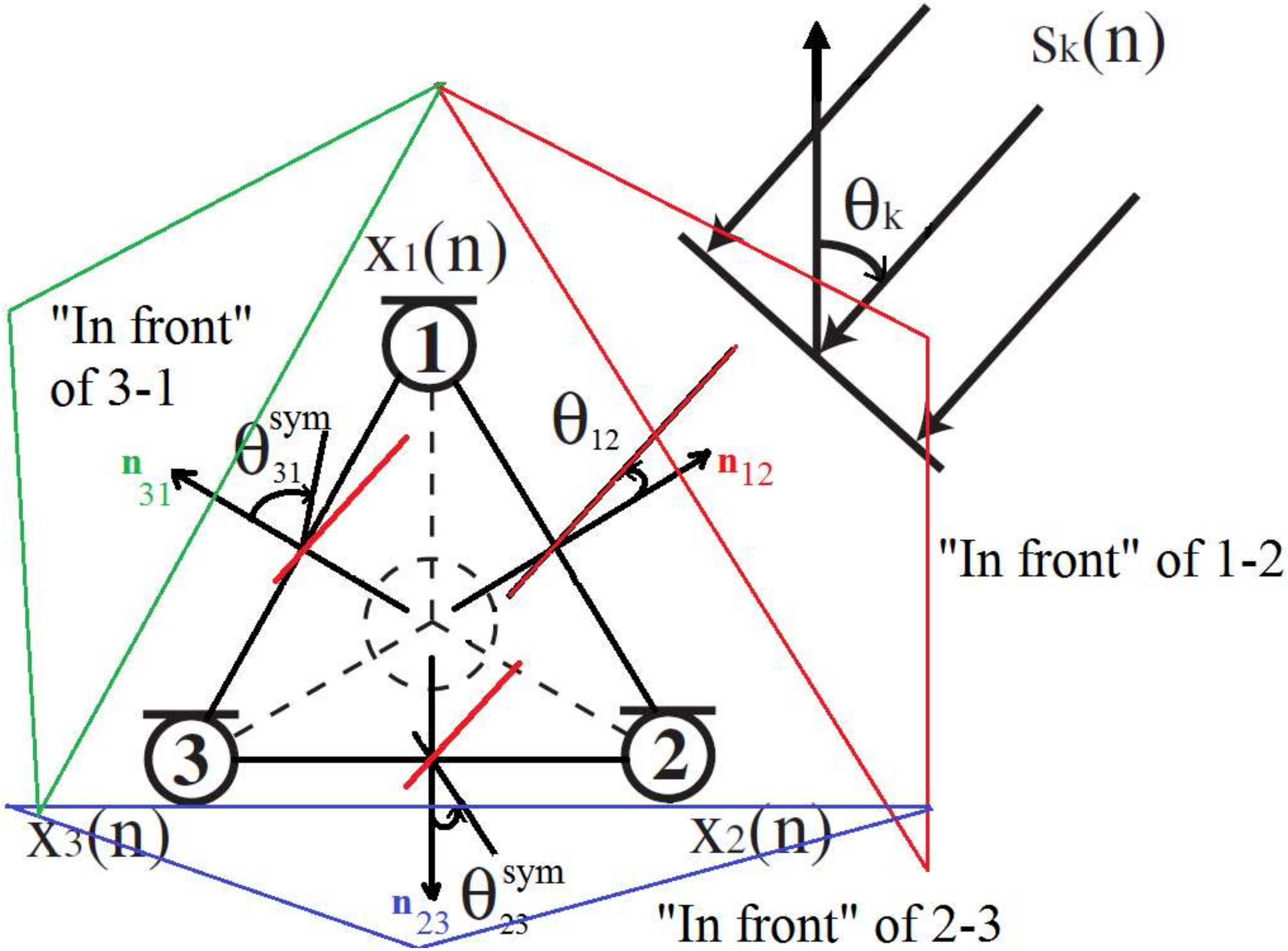
To achieve omni-directional source detection and to disambiguate the individual measurements – use a triangle of microphones:

d= 4 ÷ 8 cm



Combination of 3 detection pairs

Location „in front" of one pair corresponds to two „symmetric" directions for the two remaining pairs:



Orientation voting method

1. For every microphone pair

   For every histogram peak

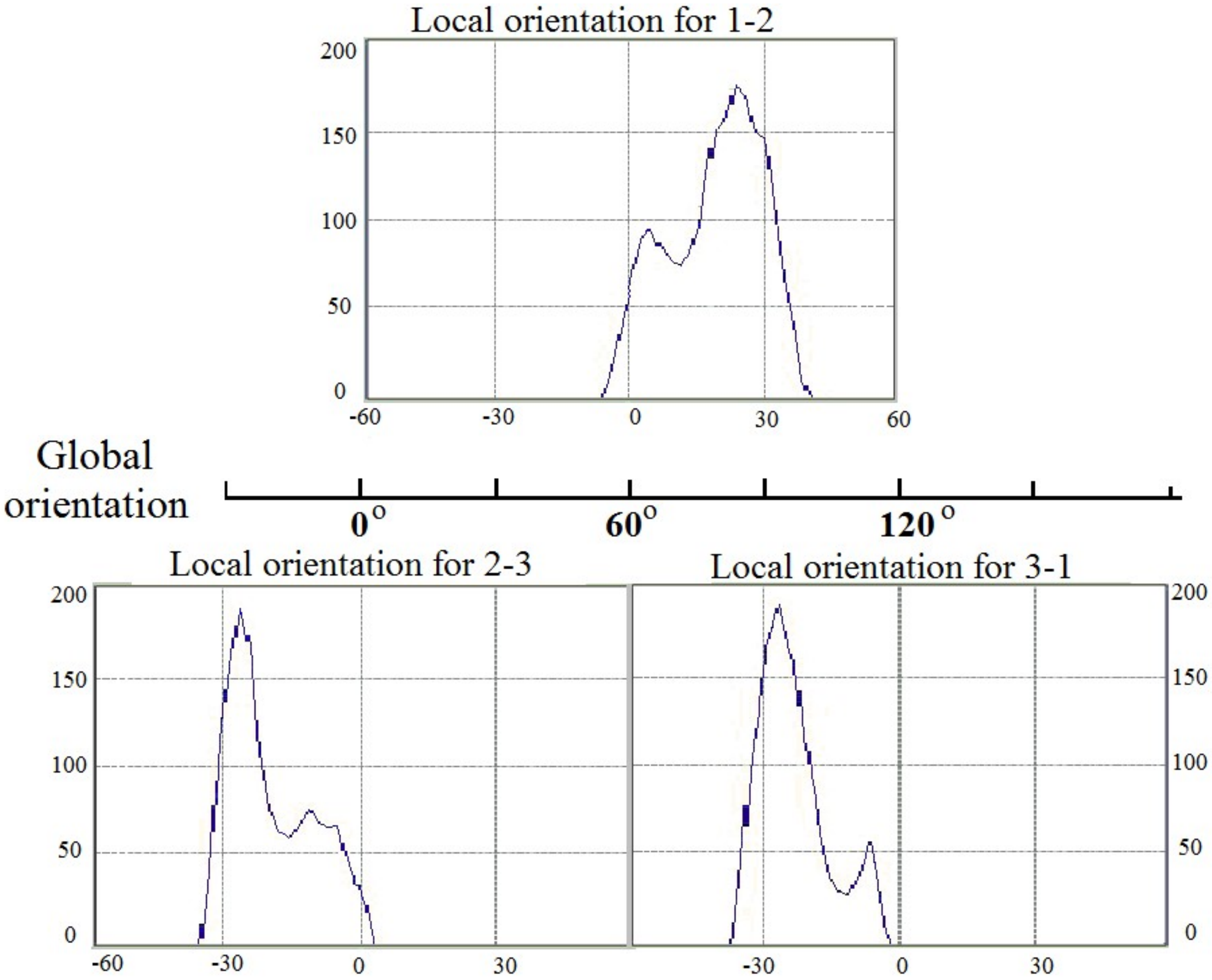
   add orientation value

   add „symmetric” orientation value

2. Select orientations with highest number of votes

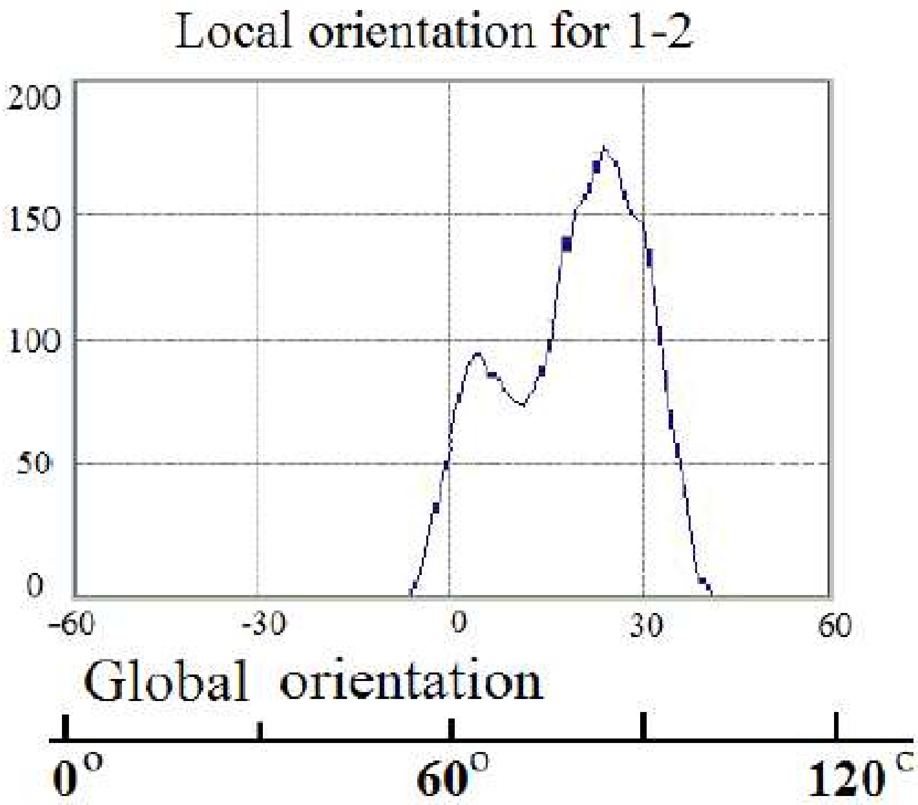
Example: restricted orientation

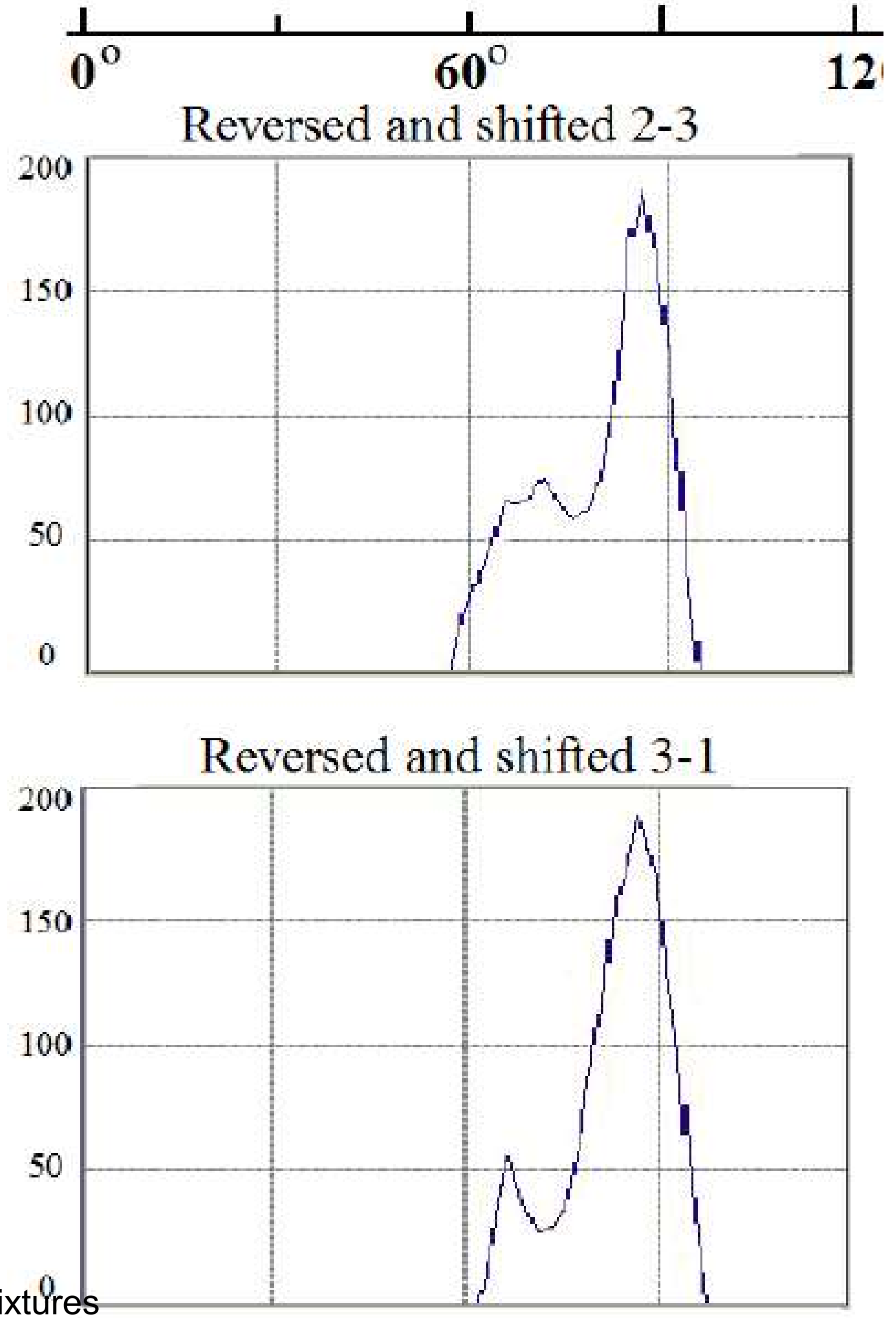
Assume the source location is restricted to the front of pair 1-2 (orientation ambiguity is cancelled))



Example (2)

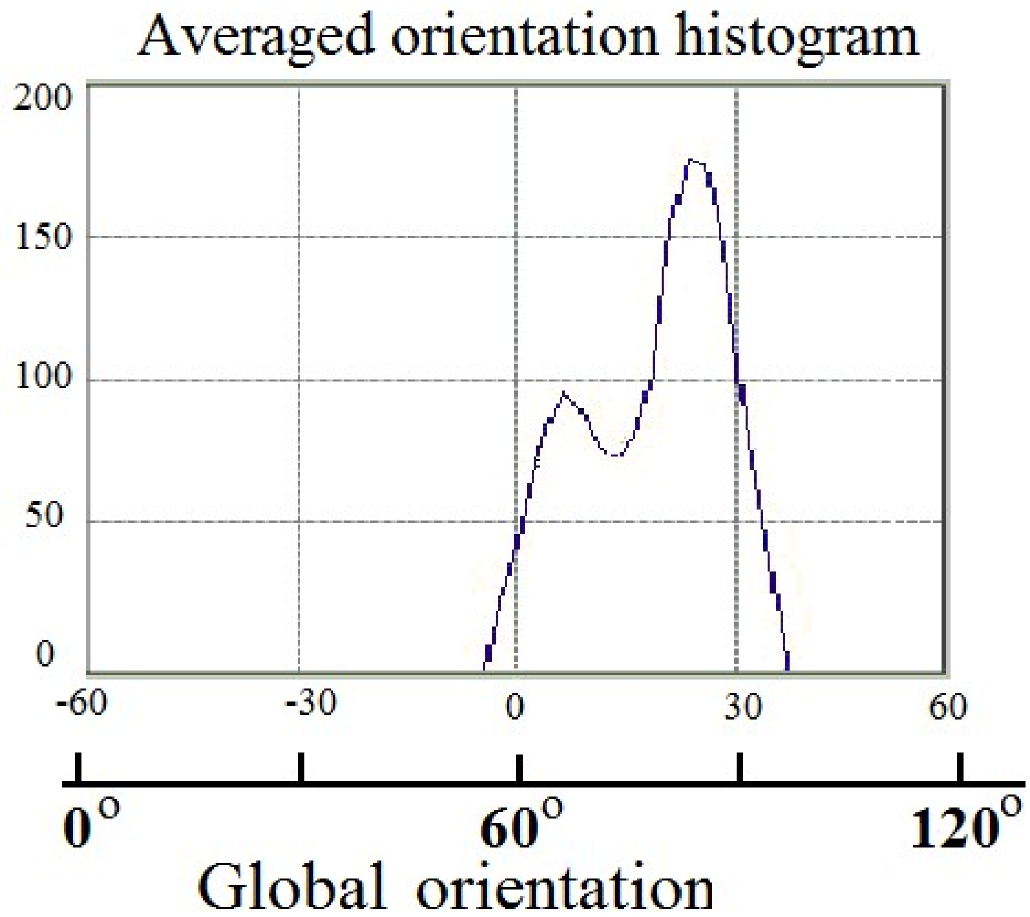
Local-to-global direction mapping:





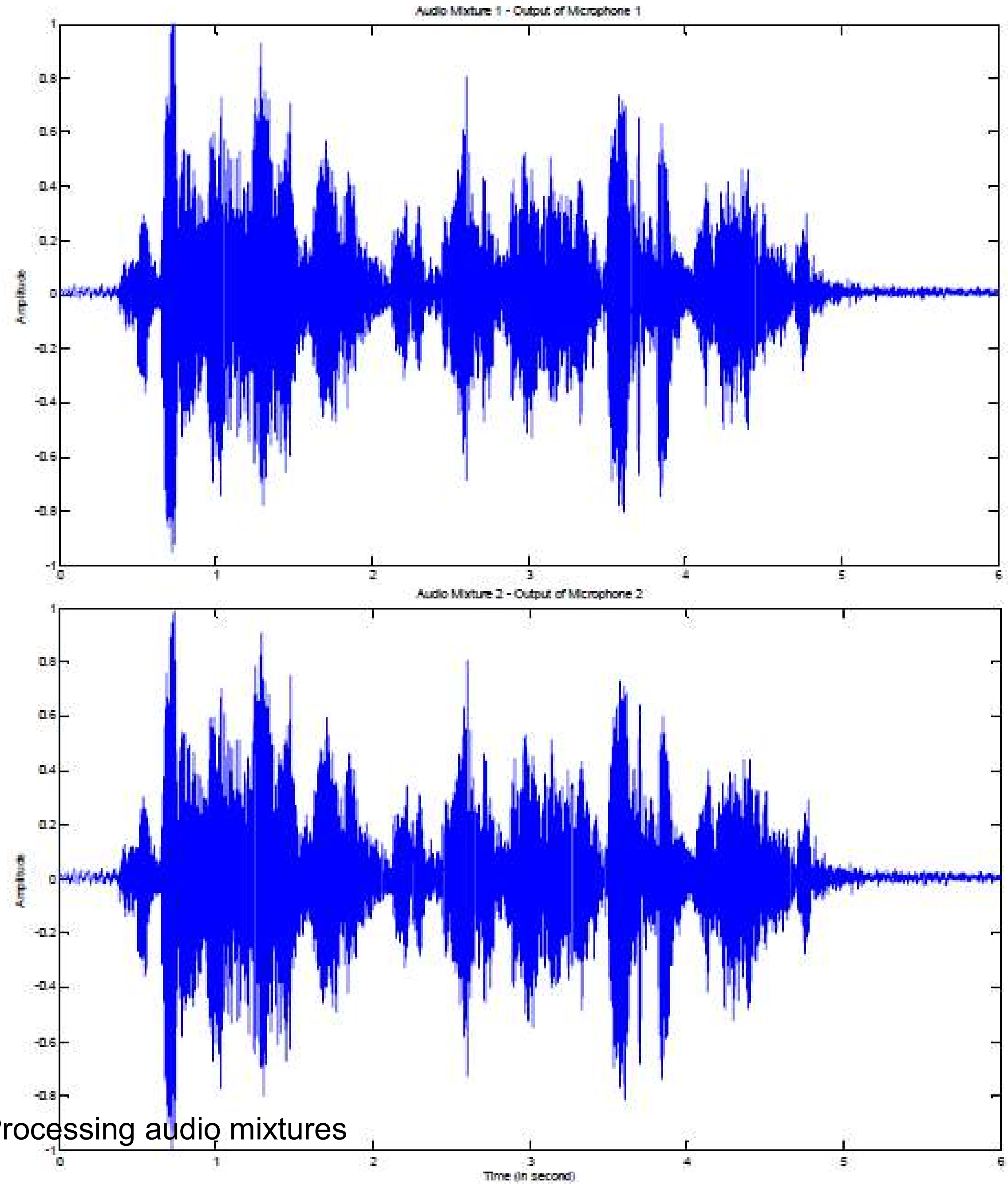
Example (3)

The three histograms can now be summarized and the peaks be detected:



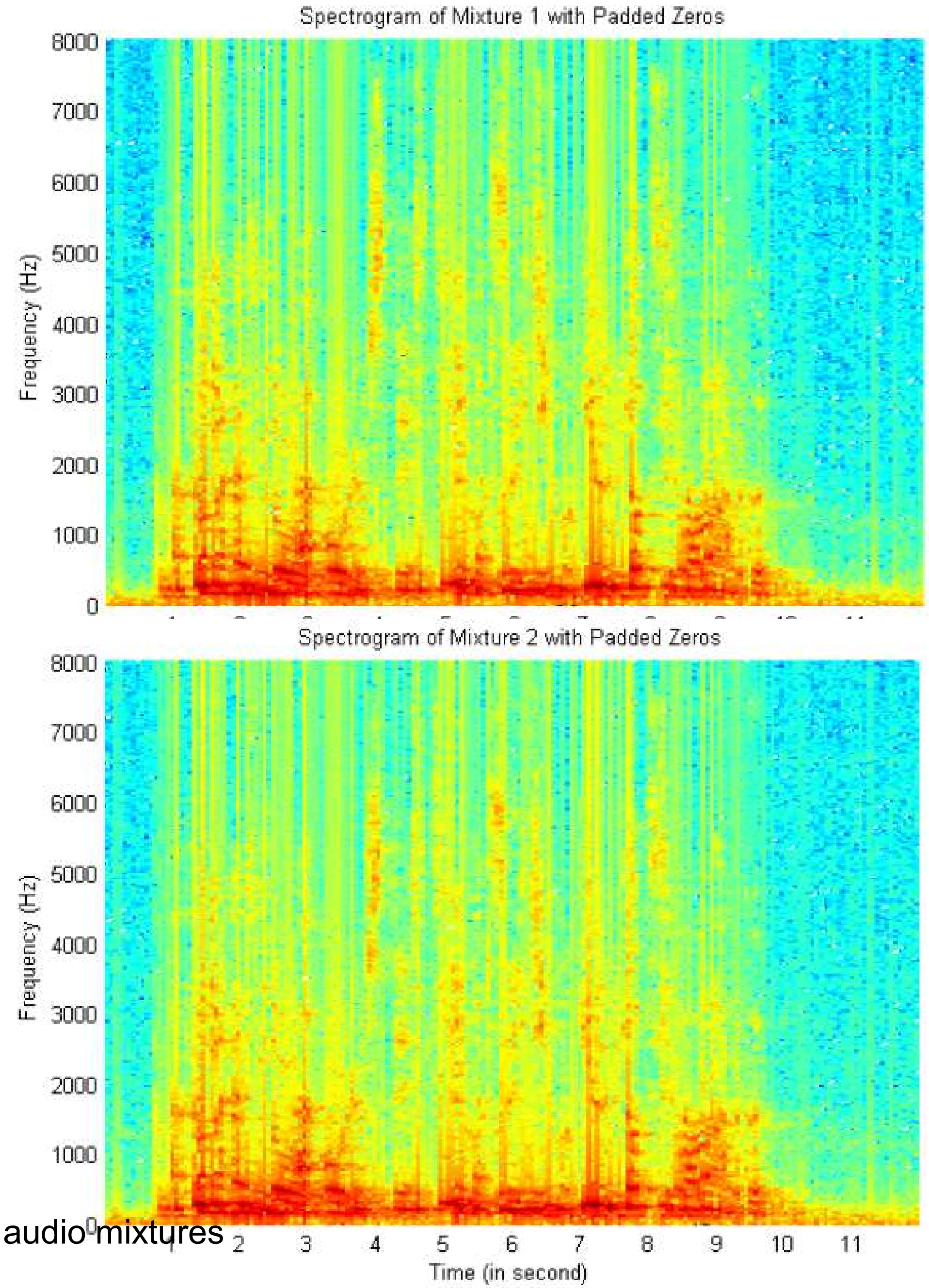
4. Source extraction

In time domain the two sources are nearly the same

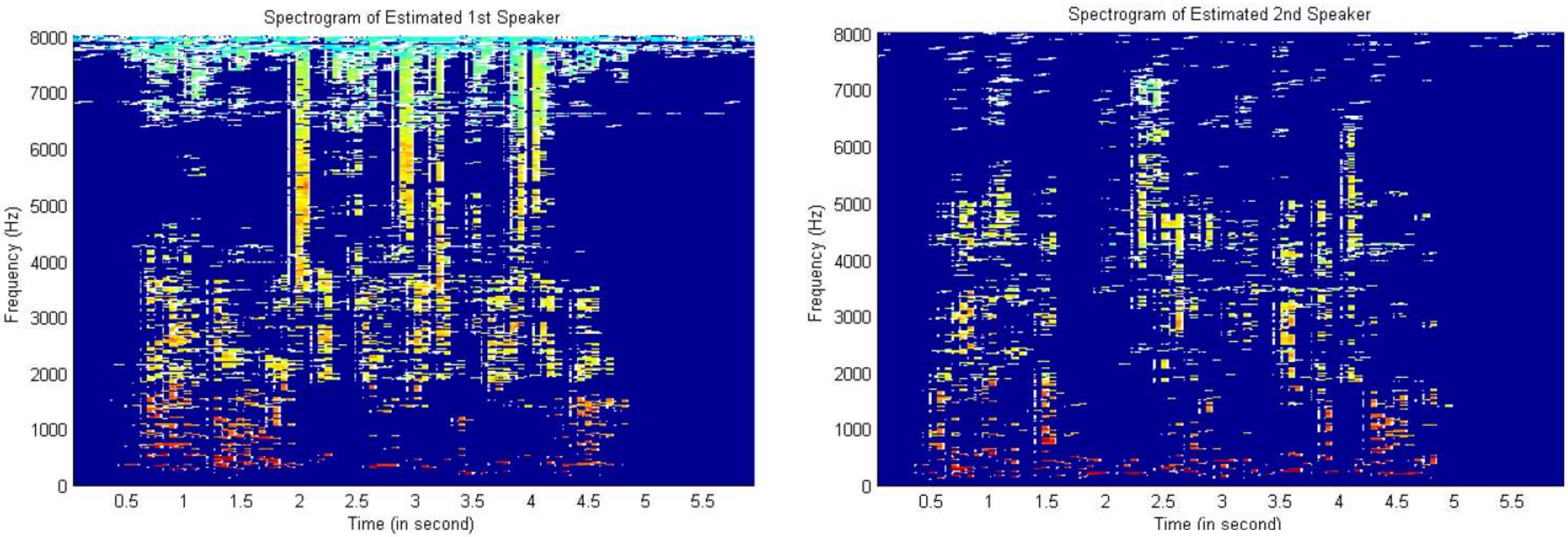


Example: two spectrograms

At frequencies from 0 to 4000 Hz the amplitude spectrograms are the same



Complementary spectrograms



Main drawback of TF-masking (with binary masks): it gives highly sparse representation of separated sources.

Proposed improvement: fractional weights for frequency bins in which more than one source is active.

Multi-valued mask

Time-frequency masking by a multi-valued mask.
The separation quality is given by a WDO factor:

$$WDO = \frac{||M(t,f)S_d(t,f)||^2 - ||M(t,f)S_i(t,f)||^2}{||S_d(t,f)||^2}$$

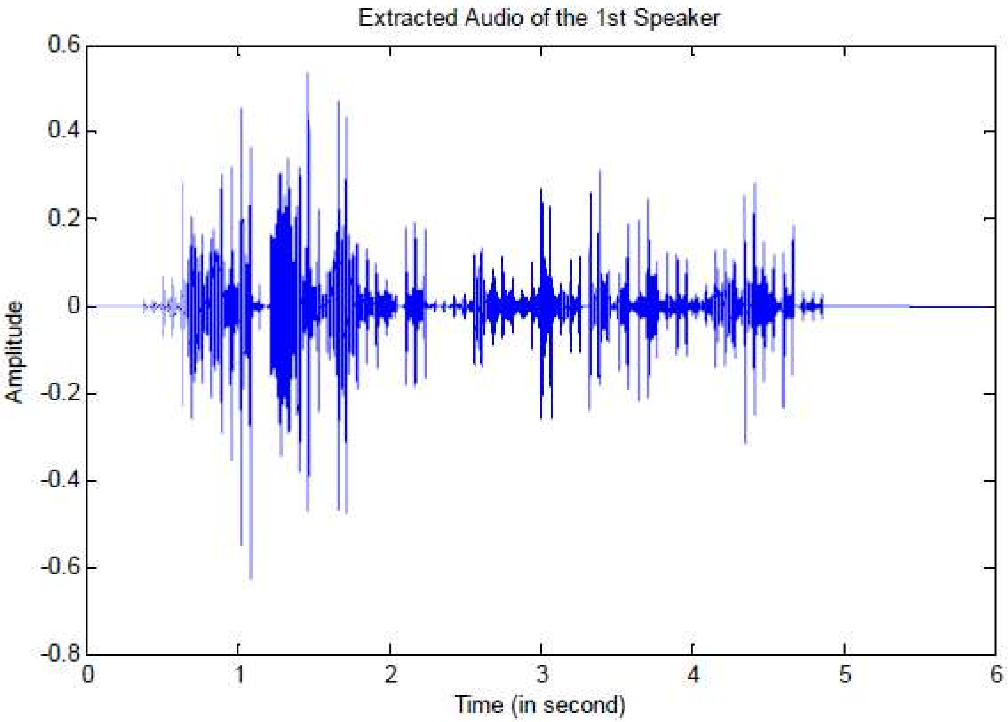
where $M()$ – mask for given source S_d, S_d – destination source, S_i – interference.

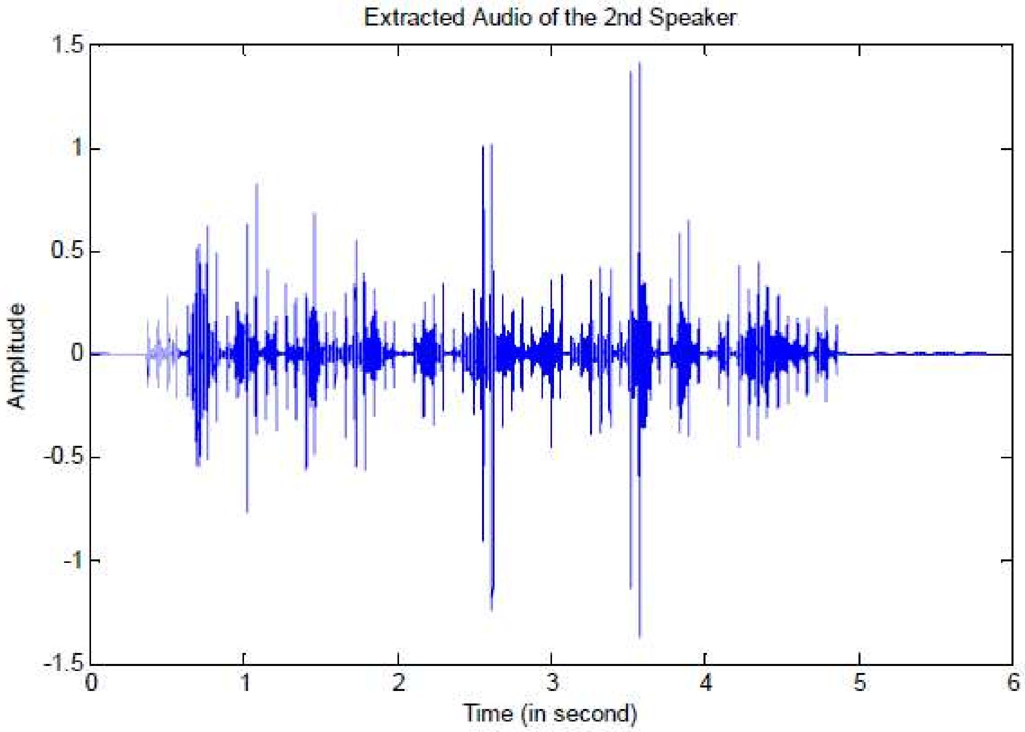
$0 \leq \text{WDO} \leq 1$. For ideal separation: WDO=1.

Most difficult cases:

Women at: Men at	60^o	70^o	80^o	90^o
50^o	0.9264	0.9109	0.9004	0.8808
60^o		0.9073	0.9020	0.8807
70^o			0.8166	0.6876
80^o				0.4491

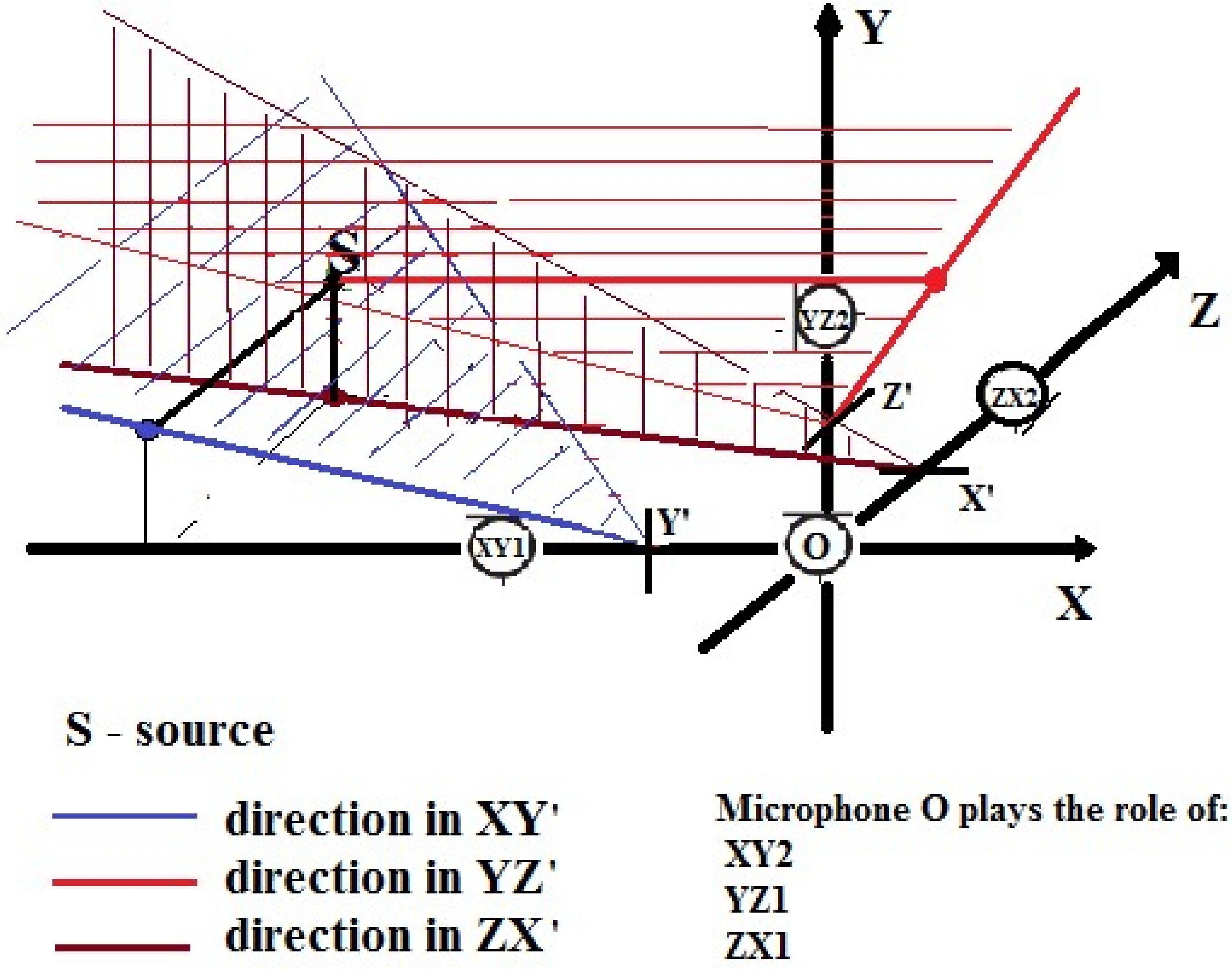
Example: extracted sources





5. 3D localization

A minimum configuration of 4 microphones:



Limited distance

Due to finite orientation step, for sufficiently far located sources individual planes will be nearly parallel.

Largest allowed distance:

$$r_{max} = \frac{\pi d}{\alpha}$$

where d – base distance, α - orientation step.

E.g. d= 8 cm, α = 10^{o} : r $_{max}$ = 1.44 m.

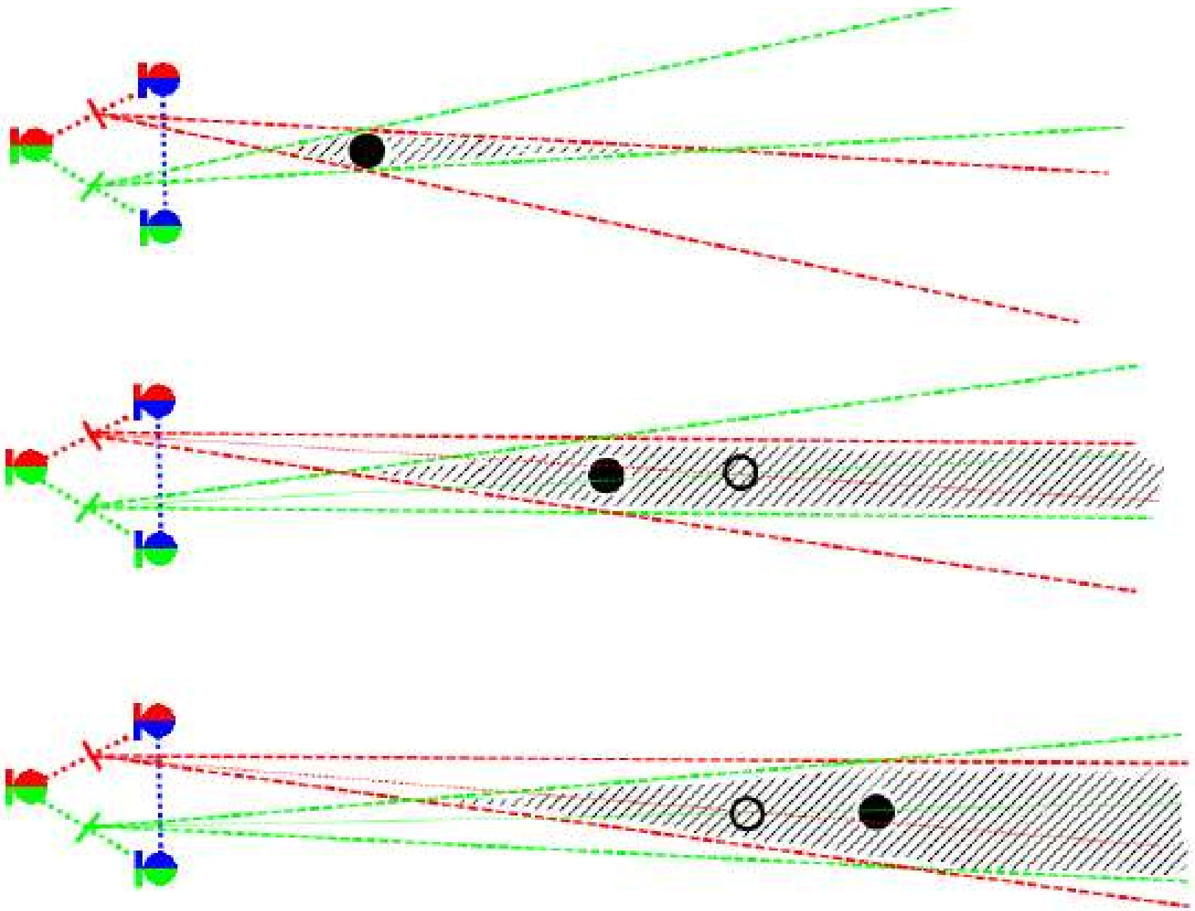
Solution: use redundant systems of 4 microphones.

Limited distance - illustration

Proper distance

Border distance

Distance exceeded



6. Summary

TDA –based audio analysis:

- Passive system – only signal acquisition
- Small base distance of microphones (4-8 cm)
- Many-source separation due to the WDO assumption
- A pair of microphones – basic element of 2D localization
- Triangle of microphones - for omni directions
- Quadruple of microphones – basic 3D localization

III. Exercises

Task 6.1

Simulate the AMUSE algorithm for the following sources:

$\boldsymbol{S}$ \ n	0	1	2	3	4	5
S_1	-1	0	1	0	-1	0
S_2	-1	1	1	-1	-1	1

and mixing matrix: $\boldsymbol{A} = \begin{bmatrix} 1 & 1 \\ -1 & 2 \end{bmatrix}$.

Get the mixed signals and estimate the mixing matrix.

Task 6.2

Simulate the on-line NG BSS algorithm assuming the nonlinearities: $f\left(y(k)\right) = y(k)^3; g(y(k)) = y(k)$

Assume following sources and mixing matrix:

S \ n	0	1	2	3	4	5
S1	-1	0	1	0	-1	1
S2	-1	1	1	-1	-1	1

$$\boldsymbol{A} = \begin{bmatrix} 1 & 1 \\ -1 & 2 \end{bmatrix}$$

Initialization of parameters:

$$\rho(0) = 0.1, \qquad \boldsymbol{W}(0) = \begin{bmatrix} 0.1 & 0.1 \\ -0.2 & 0.1 \end{bmatrix}$$

Obtain first 3 weights and outputs of the demixing network:
$\boldsymbol{W}(1)$, $\boldsymbol{W}(2)$, $\boldsymbol{W}(3)$ and $\boldsymbol{y}(1)$, $\boldsymbol{y}(2)$, $\boldsymbol{y}(3)$

Task 6.3

(ICA) The following 3 discrete-time source signals are available,
S1 = "Triangle signal", S2 =„Rectangle signal", S3="Noise" :

source \ n	0	1	2	3	4	5	6	7		
S1	1	2	1	0	-1	-2	-1	0		
S2	1	1	-1	-1	1	1	-1	-1		
S3	2	-1	1	-2	0	-1	-1	2		

(1) Compute the **normalized kurtosis** of each source signal.
(2) Make 3 **instantaneous mixtures** of sources using the matrix:

$$A_{3\times 3} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 2 & 1 \\ 2 & 2 & 1 \end{bmatrix}$$

(3) Compute the normalized **correlation factor** of pairs of sources and pairs of mixtures.

Task 6.4

Assume a (usually unknown) mixing matrix: $\boldsymbol{A}_{3\times 3} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 2 & 1 \\ 2 & 2 & 1 \end{bmatrix}$

and a final de-mixing matrix $\boldsymbol{W}$:

$$\boldsymbol{W}_{3\times 3} = \begin{bmatrix} 2.2 & 0.1 & -1 \\ 0.1 & -1 & 1 \\ -1.1 & 1 & 0.1 \end{bmatrix}$$

Calculate the **error index of separation** $EI(\boldsymbol{P})$.