Blind Source Separation Methods Based on Approximate Joint Diagonalization Algorithms

Petr Tichavský, Tomáš Marek

Institute of Information Theory and Automation,
Academy of Sciences of the Czech Republic

The Linear and Instantaneous BSS Mixing Model

The model can be formulated as

$$\mathbf{x} = \mathbf{A}\mathbf{s}$$

$\mathbf{s}$	$d \times N$ data matrix having as rows the unobserved source signals $\mathbf{s}_k$, $k = 1, \ldots, d$
$\mathbf{A}$	An unknown $d \times d$ regular mixing matrix
$\mathbf{x}$	The mixed (observed) signals

The goal is to estimate a separating matrix $\widehat{\mathbf{W}}$ such that

$$\hat{\mathbf{s}} = \widehat{\mathbf{W}}\mathbf{x} \approx \mathbf{s}$$

(we assume that correct scales and order of the sources were recovered)

The Linear and Instantaneous BSS Mixing Model

The model can be formulated as

$$\mathbf{x} = \mathbf{A}\mathbf{s}$$

$\mathbf{s}$	$d \times N$ data matrix having as rows the unobserved source signals $\mathbf{s}_k$, $k = 1, \ldots, d$
$\mathbf{A}$	An unknown $d \times d$ regular mixing matrix
$\mathbf{x}$	The mixed (observed) signals

The goal is to estimate a separating matrix $\widehat{\mathbf{W}}$ such that

$$\hat{\mathbf{s}} = \widehat{\mathbf{W}}\mathbf{x} \approx \mathbf{s}$$

(we assume that correct scales and order of the sources were recovered)

The Linear and Instantaneous BSS Mixing Model

The model can be formulated as

$$\mathbf{x} = \mathbf{A}\mathbf{s}$$

$\mathbf{s}$	$d \times N$ data matrix having as rows the unobserved source signals $\mathbf{s}_k$, $k = 1, \ldots, d$
$\mathbf{A}$	An unknown $d \times d$ regular mixing matrix
$\mathbf{x}$	The mixed (observed) signals

The goal is to estimate a separating matrix $\widehat{\mathbf{W}}$ such that

$$\hat{\mathbf{s}} = \widehat{\mathbf{W}}\mathbf{x} \approx \mathbf{s}$$

(we assume that correct scales and order of the sources were recovered)

Performance Measure: the ISR Matrix

If the sources are well estimated then $\hat{\mathbf{s}} \approx \mathbf{s}$ and

$$\mathbf{G} = \widehat{\mathbf{V}}\mathbf{A} \approx \mathbf{I}$$

The Interference-to-Signal Ratio (ISR) matrix:

$$\mathbf{ISR}_{k\ell} = \frac{\mathbf{G}_{k\ell}^2}{\mathbf{G}_{kk}^2}, \quad k, \ell = 1, 2, ..., d$$

The ISR of the k-th estimated signal is the k-th element of a d-dimensional vector **isr**:

$$\mathbf{isr}_k = \frac{\sum_{\ell=1,\ell\neq k}^{d} \mathbf{G}_{k\ell}^2}{\mathbf{G}_{kk}^2} \quad k = 1, 2, ..., d$$

Performance Measure: the ISR Matrix

If the sources are well estimated then $\hat{\mathbf{s}} \approx \mathbf{s}$ and

$$\mathbf{G} = \widehat{\mathbf{V}}\mathbf{A} \approx \mathbf{I}$$

The Interference-to-Signal Ratio (ISR) matrix:

$$\mathbf{ISR}_{k\ell} = \frac{\mathbf{G}_{k\ell}^2}{\mathbf{G}_{kk}^2}, \quad k, \ell = 1, 2, ..., d$$

The ISR of the k-th estimated signal is the k-th element of a d-dimensional vector **isr**:

$$\mathbf{isr}_k = \frac{\sum_{\ell=1,\ell\neq k}^{d} \mathbf{G}_{k\ell}^2}{\mathbf{G}_{kk}^2} \quad k = 1, 2, ..., d$$

Performance Measure: the ISR Matrix

If the sources are well estimated then $\hat{\mathbf{s}} \approx \mathbf{s}$ and

$$\mathbf{G} = \widehat{\mathbf{V}}\mathbf{A} \approx \mathbf{I}$$

The Interference-to-Signal Ratio (ISR) matrix:

$$\mathbf{ISR}_{k\ell} = \frac{\mathbf{G}_{k\ell}^2}{\mathbf{G}_{kk}^2}, \quad k, \ell = 1, 2, ..., d$$

The ISR of the k-th estimated signal is the k-th element of a d-dimensional vector **isr**:

$$\mathbf{isr}_k = \frac{\sum_{\ell=1,\ell\neq k}^{d} \mathbf{G}_{k\ell}^2}{\mathbf{G}_{kk}^2} \quad k = 1, 2, ..., d$$

Source signals' models / Algorithms

1. Non-Gaussian independent and identically distributed (i.i.d.) processes (JADE, FastICA, Infomax, ..., EFICA)
2. Weakly stationary random processes driven by white Gaussian noise, (SOBI, WASOBI)
3. Sequences of independent Gaussian variables with time-varying variances (Block Gaussian Likelihood by Pham)
4. Combinations of the three models above (JADE_{TD}, ThinICA, Unified method by Hyvärinen, COMBI, MultiCOMBI, Pham's method ...)
5. Sources with bounded support / nonnegative sources (Erdogan, Vrins, Plumbley)

Source signals' models / Algorithms

1. Non-Gaussian independent and identically distributed (i.i.d.) processes (JADE, FastICA, Infomax, ..., EFICA)
2. Weakly stationary random processes driven by white Gaussian noise, (SOBI, WASOBI)
3. Sequences of independent Gaussian variables with time-varying variances (Block Gaussian Likelihood by Pham)
4. Combinations of the three models above (JADE_{TD}, ThinICA, Unified method by Hyvärinen, COMBI, MultiCOMBI, Pham's method ...)
5. Sources with bounded support / nonnegative sources (Erdogan, Vrins, Plumbley)

Source signals' models / Algorithms

1. Non-Gaussian independent and identically distributed (i.i.d.) processes (JADE, FastICA, Infomax, ..., EFICA)
2. Weakly stationary random processes driven by white Gaussian noise, (SOBI, WASOBI)
3. Sequences of independent Gaussian variables with time-varying variances (Block Gaussian Likelihood by Pham)
4. Combinations of the three models above (JADE_{TD}, ThinICA, Unified method by Hyvärinen, COMBI, MultiCOMBI, Pham's method ...)
5. Sources with bounded support / nonnegative sources (Erdogan, Vrins, Plumbley)

Source signals' models / Algorithms

1. Non-Gaussian independent and identically distributed (i.i.d.) processes (JADE, FastICA, Infomax, ..., EFICA)
2. Weakly stationary random processes driven by white Gaussian noise, (SOBI, WASOBI)
3. Sequences of independent Gaussian variables with time-varying variances (Block Gaussian Likelihood by Pham)
4. Combinations of the three models above (JADE_{TD}, ThinICA, Unified method by Hyvärinen, COMBI, MultiCOMBI, Pham's method ...)
5. Sources with bounded support / nonnegative sources (Erdogan, Vrins, Plumbley)

Source signals' models / Algorithms

1. Non-Gaussian independent and identically distributed (i.i.d.) processes (JADE, FastICA, Infomax, ..., EFICA)
2. Weakly stationary random processes driven by white Gaussian noise, (SOBI, WASOBI)
3. Sequences of independent Gaussian variables with time-varying variances (Block Gaussian Likelihood by Pham)
4. Combinations of the three models above (JADE_{TD}, ThinICA, Unified method by Hyvärinen, COMBI, MultiCOMBI, Pham's method ...)
5. Sources with bounded support / nonnegative sources (Erdogan, Vrins, Plumbley)

Algorithm WASOBI

- SOS-based BSS algorithm based on assumption of spectral diversity of the sources.
- The mixing matrix is estimated through the relation of the time-lagged sample correlation matrices of the observed mixtures and the original sources:

$$\mathbf{R_x}[\tau] = \frac{1}{N}\sum_{n=1}^{N}\mathbf{x}[n]\mathbf{x}^T[n+\tau] = \mathbf{A}\mathbf{R_s}[\tau]\mathbf{A}^T \qquad \tau = 0,\ldots,M-1$$

- Optimizes SOBI (asymptotically, for Gaussian sources) by reformulating the joint diagonalization (AJD) problem as a non-linear, optimally weighted least squares problem.

Algorithm WASOBI

- SOS-based BSS algorithm based on assumption of spectral diversity of the sources.
- The mixing matrix is estimated through the relation of the time-lagged sample correlation matrices of the observed mixtures and the original sources:

$$\mathbf{R_x}[\tau] = \frac{1}{N}\sum_{n=1}^{N} \mathbf{x}[n]\mathbf{x}^T[n+\tau] = \mathbf{A}\mathbf{R_s}[\tau]\mathbf{A}^T \qquad \tau = 0, \ldots, M-1$$

- Optimizes SOBI (asymptotically, for Gaussian sources) by reformulating the joint diagonalization (AJD) problem as a non-linear, optimally weighted least squares problem.

Algorithm WASOBI

- SOS-based BSS algorithm based on assumption of spectral diversity of the sources.
- The mixing matrix is estimated through the relation of the time-lagged sample correlation matrices of the observed mixtures and the original sources:

$$\mathbf{R_x}[\tau] = \frac{1}{N}\sum_{n=1}^{N}\mathbf{x}[n]\mathbf{x}^T[n+\tau] = \mathbf{A}\mathbf{R_s}[\tau]\mathbf{A}^T \qquad \tau = 0, \ldots, M-1$$

- Optimizes SOBI (asymptotically, for Gaussian sources) by reformulating the joint diagonalization (AJD) problem as a non-linear, optimally weighted least squares problem.

Algorithm BGL (Block Gaussian Likelihood)

- SOS-based BSS algorithm that exploits the nonstationarity of the sources.
- Assumes that sources are i.i.d. Gaussian stationary inside blocks but with varying variance between blocks.
- The mixing matrix is estimated through the relation of the sample correlation matrices of the observed mixtures and the original sources:

$$\mathbf{R}_{\mathbf{x}}^{(\ell)} = \frac{1}{N_L} \sum_{n=(\ell-1)N_L+1}^{\ell N_L} \mathbf{x}[n]\mathbf{x}^T[n] = \mathbf{A}\mathbf{R}_{\mathbf{s}}{}^{(\ell)}\mathbf{A}^T, \qquad \ell = 1, \ldots, L$$

- BGL realizes the maximum likelihood estimator
- BGL can be alternatively implemented as WASOBI with proper weights (increased speed and stability).

Algorithm BGL (Block Gaussian Likelihood)

- SOS-based BSS algorithm that exploits the nonstationarity of the sources.
- Assumes that sources are i.i.d. Gaussian stationary inside blocks but with varying variance between blocks.
- The mixing matrix is estimated through the relation of the sample correlation matrices of the observed mixtures and the original sources:

$$\mathbf{R}_{\mathbf{x}}^{(\ell)} = \frac{1}{N_L} \sum_{n=(\ell-1)N_L+1}^{\ell N_L} \mathbf{x}[n]\mathbf{x}^T[n] = \mathbf{A}\mathbf{R}_{\mathbf{s}}{}^{(\ell)}\mathbf{A}^T, \qquad \ell = 1, \ldots, L$$

- BGL realizes the maximum likelihood estimator
- BGL can be alternatively implemented as WASOBI with proper weights (increased speed and stability).

Algorithm BGL (Block Gaussian Likelihood)

- SOS-based BSS algorithm that exploits the nonstationarity of the sources.
- Assumes that sources are i.i.d. Gaussian stationary inside blocks but with varying variance between blocks.
- The mixing matrix is estimated through the relation of the sample correlation matrices of the observed mixtures and the original sources:

$$\mathbf{R}_{\mathbf{x}}^{(\ell)} = \frac{1}{N_L} \sum_{n=(\ell-1)N_L+1}^{\ell N_L} \mathbf{x}[n]\mathbf{x}^T[n] = \mathbf{A}\mathbf{R_s}^{(\ell)}\mathbf{A}^T, \qquad \ell = 1, \dots, L$$

- BGL realizes the maximum likelihood estimator
- BGL can be alternatively implemented as WASOBI with proper weights (increased speed and stability).

Algorithm BGL (Block Gaussian Likelihood)

- SOS-based BSS algorithm that exploits the nonstationarity of the sources.
- Assumes that sources are i.i.d. Gaussian stationary inside blocks but with varying variance between blocks.
- The mixing matrix is estimated through the relation of the sample correlation matrices of the observed mixtures and the original sources:

$$\mathbf{R}_{\mathbf{x}}^{(\ell)} = \frac{1}{N_L} \sum_{n=(\ell-1)N_L+1}^{\ell N_L} \mathbf{x}[n]\mathbf{x}^T[n] = \mathbf{A}\mathbf{R}_{\mathbf{s}}{}^{(\ell)}\mathbf{A}^T, \qquad \ell = 1, \ldots, L$$

- BGL realizes the maximum likelihood estimator
- BGL can be alternatively implemented as WASOBI with proper weights (increased speed and stability).

Algorithm BGL (Block Gaussian Likelihood)

- SOS-based BSS algorithm that exploits the nonstationarity of the sources.
- Assumes that sources are i.i.d. Gaussian stationary inside blocks but with varying variance between blocks.
- The mixing matrix is estimated through the relation of the sample correlation matrices of the observed mixtures and the original sources:

$$\mathbf{R}_{\mathbf{x}}^{(\ell)} = \frac{1}{N_L} \sum_{n=(\ell-1)N_L+1}^{\ell N_L} \mathbf{x}[n]\mathbf{x}^T[n] = \mathbf{A}\mathbf{R}_{\mathbf{s}}{}^{(\ell)}\mathbf{A}^T, \qquad \ell = 1, \ldots, L$$

- BGL realizes the maximum likelihood estimator
- BGL can be alternatively implemented as WASOBI with proper weights (increased speed and stability).

Approximate Joint Diagonalization (AJD)

Input: A set of square ($d \times d$) matrices

$$\mathbf{R_x}[\tau], \tau = 1, \dots, M$$

Desired output: A square ($d \times d$) matrix $\mathbf{V}$ such that

$$\mathbf{V}\mathbf{R_x}[\tau]\mathbf{V}^T, \tau = 1, \dots, M$$

are "as much diagonal as possible".

In the case $M = 2$ an exact joint diagonalization is possible. Here $\mathbf{V}$ is composed of generalized eigenvectors of the matrix pencil $(\mathbf{R_x}[1], \mathbf{R_x}[2])$, i.e. solutions of $\mathbf{R_x}[1]\mathbf{x} = \lambda\mathbf{R_x}[2]\mathbf{x}$.

Approximate Joint Diagonalization (AJD)

Input: A set of square ($d \times d$) matrices

$$\mathbf{R_x}[\tau], \tau = 1, \ldots, M$$

Desired output: A square ($d \times d$) matrix $\mathbf{V}$ such that

$$\mathbf{V}\mathbf{R_x}[\tau]\mathbf{V}^T, \tau = 1, \ldots, M$$

are "as much diagonal as possible".

In the case $M = 2$ an exact joint diagonalization is possible. Here $\mathbf{V}$ is composed of generalized eigenvectors of the matrix pencil $(\mathbf{R_x}[1], \mathbf{R_x}[2])$, i.e. solutions of $\mathbf{R_x}[1]\mathbf{x} = \lambda\mathbf{R_x}[2]\mathbf{x}$.

AJD algorithms for WASOBI and BGL

- LLAJD (Pham) $\min_{\mathbf{V}} \sum_{\tau=1}^{M} \log \frac{\det \operatorname{diag}(\mathbf{V}\mathbf{R}_{\mathbf{x}}[\tau]\mathbf{V}^T)}{\det(\mathbf{V}\mathbf{R}_{\mathbf{x}}[\tau]\mathbf{V}^T)}$
- ACDC (Yeredor) $\min_{\mathbf{V},\mathbf{D}_\tau} \sum_{\tau=0}^{M-1} \|\mathbf{R}_{\mathbf{x}}[\tau] - \mathbf{V}^{-1}\mathbf{D}_\tau\mathbf{V}^{-T}\|_F^2$
- FAJD (Li, Zhang) $\min_{\mathbf{V}} \sum_{\tau=1}^{M} \|\operatorname{off}(\mathbf{V}\mathbf{R}_{\mathbf{x}}[\tau]\mathbf{V}^T)\|_F^2 - \beta \log|\det \mathbf{V}|$
- FFDIAG (Ziehe et al) ... the same as UWAJD ?
- QAJD (Vollgraf, Obermayer) $\min_{\mathbf{V}} \sum_{\tau=1}^{M} \|\operatorname{off}(\mathbf{V}\mathbf{R}_{\mathbf{x}}[\tau]\mathbf{V}^T)\|_F^2$ subj. to $\operatorname{diag}(\mathbf{V}\mathbf{R}_{\mathbf{x}}[0]\mathbf{V}^T) = \operatorname{ones}(d,1)$
- UWAJD (icassp 2008): $\mathbf{V}$ is solution to $\varphi(\mathbf{V}) = \mathbf{I}$, $\operatorname{diag}(\mathbf{V}\mathbf{R}_{\mathbf{x}}[0]\mathbf{V}^T) = \operatorname{ones}(d,1)$, where $\varphi(\mathbf{V}) = \operatorname{argmin}_{\mathbf{A}} \sum_{\tau=1}^{M} \|\mathbf{V}\mathbf{R}_{\mathbf{x}}[\tau]\mathbf{V}^T - \mathbf{A}\mathbf{D}_{\tau,\mathbf{V}}\mathbf{A}^T\|_F^2$
  $\mathbf{D}_{\tau,\mathbf{V}} = \operatorname{diag}(\operatorname{diag}(\mathbf{V}\mathbf{R}_{\mathbf{x}}[\tau]\mathbf{V}^T))$

AJD algorithms for WASOBI and BGL

- LLAJD (Pham) $\min_{\mathbf{V}} \sum_{\tau=1}^{M} \log \frac{\det \operatorname{diag}(\mathbf{V}\mathbf{R}_{\mathbf{x}}[\tau]\mathbf{V}^T)}{\det(\mathbf{V}\mathbf{R}_{\mathbf{x}}[\tau]\mathbf{V}^T)}$
- ACDC (Yeredor) $\min_{\mathbf{V},\mathbf{D}_\tau} \sum_{\tau=0}^{M-1} \|\mathbf{R}_{\mathbf{x}}[\tau] - \mathbf{V}^{-1}\mathbf{D}_\tau\mathbf{V}^{-T}\|_{\mathrm{F}}^2$
- FAJD (Li, Zhang)
  $\min_{\mathbf{V}} \sum_{\tau=1}^{M} \|\operatorname{off}(\mathbf{V}\mathbf{R}_{\mathbf{x}}[\tau]\mathbf{V}^T)\|_{\mathrm{F}}^2 - \beta \log|\det \mathbf{V}|$
- FFDIAG (Ziehe et al) ... the same as UWAJD ?
- QAJD (Vollgraf, Obermayer) $\min_{\mathbf{V}} \sum_{\tau=1}^{M} \|\operatorname{off}(\mathbf{V}\mathbf{R}_{\mathbf{x}}[\tau]\mathbf{V}^T)\|_{\mathrm{F}}^2$
  subj. to $\operatorname{diag}(\mathbf{V}\mathbf{R}_{\mathbf{x}}[0]\mathbf{V}^T) = \operatorname{ones}(d, 1)$
- UWAJD (icassp 2008): $\mathbf{V}$ is solution to $\varphi(\mathbf{V}) = \mathbf{I}$,
  $\operatorname{diag}(\mathbf{V}\mathbf{R}_{\mathbf{x}}[0]\mathbf{V}^T) = \operatorname{ones}(d, 1)$, where
  $\varphi(\mathbf{V}) = \operatorname{argmin}_{\mathbf{A}} \sum_{\tau=1}^{M} \|\mathbf{V}\mathbf{R}_{\mathbf{x}}[\tau]\mathbf{V}^T - \mathbf{A}\mathbf{D}_{\tau,\mathbf{V}}\mathbf{A}^T\|_{\mathrm{F}}^2$
  $\mathbf{D}_{\tau,\mathbf{V}} = \operatorname{diag}(\operatorname{diag}(\mathbf{V}\mathbf{R}_{\mathbf{x}}[\tau]\mathbf{V}^T))$

AJD algorithms for WASOBI and BGL

- LLAJD (Pham) $\min_{\mathbf{V}} \sum_{\tau=1}^{M} \log \frac{\det\ \mathrm{diag}(\mathbf{V}\mathbf{R}_{\mathbf{x}}[\tau]\mathbf{V}^T)}{\det(\mathbf{V}\mathbf{R}_{\mathbf{x}}[\tau]\mathbf{V}^T)}$
- ACDC (Yeredor) $\min_{\mathbf{V},\mathbf{D}_\tau} \sum_{\tau=0}^{M-1} \|\mathbf{R}_{\mathbf{x}}[\tau] - \mathbf{V}^{-1}\mathbf{D}_\tau \mathbf{V}^{-T}\|_{\mathrm{F}}^2$
- FAJD (Li, Zhang)
  $\min_{\mathbf{V}} \sum_{\tau=1}^{M} \|\mathrm{off}(\mathbf{V}\mathbf{R}_{\mathbf{x}}[\tau]\mathbf{V}^T)\|_{\mathrm{F}}^2 - \beta \log |\det \mathbf{V}|$
- FFDIAG (Ziehe et al) ... the same as UWAJD ?
- QAJD (Vollgraf, Obermayer) $\min_{\mathbf{V}} \sum_{\tau=1}^{M} \|\mathrm{off}(\mathbf{V}\mathbf{R}_{\mathbf{x}}[\tau]\mathbf{V}^T)\|_{\mathrm{F}}^2$
  subj. to $\mathrm{diag}(\mathbf{V}\mathbf{R}_{\mathbf{x}}[0]\mathbf{V}^T) = \mathrm{ones}(d,1)$
- UWAJD (icassp 2008): $\mathbf{V}$ is solution to $\varphi(\mathbf{V}) = \mathbf{I}$,
  $\mathrm{diag}(\mathbf{V}\mathbf{R}_{\mathbf{x}}[0]\mathbf{V}^T) = \mathrm{ones}(d,1)$, where
  $\varphi(\mathbf{V}) = \mathrm{argmin}_{\mathbf{A}} \sum_{\tau=1}^{M} \|\mathbf{V}\mathbf{R}_{\mathbf{x}}[\tau]\mathbf{V}^T - \mathbf{A}\mathbf{D}_{\tau,\mathbf{V}}\mathbf{A}^T\|_{\mathrm{F}}^2$
  $\mathbf{D}_{\tau,\mathbf{V}} = \mathrm{diag}(\mathrm{diag}(\mathbf{V}\mathbf{R}_{\mathbf{x}}[\tau]\mathbf{V}^T))$

AJD algorithms for WASOBI and BGL

- LLAJD (Pham) $\min_{\mathbf{V}} \sum_{\tau=1}^{M} \log \frac{\det \operatorname{diag}(\mathbf{V}\mathbf{R}_{\mathbf{x}}[\tau]\mathbf{V}^T)}{\det(\mathbf{V}\mathbf{R}_{\mathbf{x}}[\tau]\mathbf{V}^T)}$
- ACDC (Yeredor) $\min_{\mathbf{V},\mathbf{D}_\tau} \sum_{\tau=0}^{M-1} \|\mathbf{R}_{\mathbf{x}}[\tau] - \mathbf{V}^{-1}\mathbf{D}_\tau\mathbf{V}^{-T}\|_{\mathrm{F}}^2$
- FAJD (Li, Zhang)
  $\min_{\mathbf{V}} \sum_{\tau=1}^{M} \|\mathrm{off}(\mathbf{V}\mathbf{R}_{\mathbf{x}}[\tau]\mathbf{V}^T)\|_{\mathrm{F}}^2 - \beta \log|\det \mathbf{V}|$
- FFDIAG (Ziehe et al) ... the same as UWAJD ?
- QAJD (Vollgraf, Obermayer) $\min_{\mathbf{V}} \sum_{\tau=1}^{M} \|\mathrm{off}(\mathbf{V}\mathbf{R}_{\mathbf{x}}[\tau]\mathbf{V}^T)\|_{\mathrm{F}}^2$
  subj. to $\operatorname{diag}(\mathbf{V}\mathbf{R}_{\mathbf{x}}[0]\mathbf{V}^T) = \mathrm{ones}(d,1)$
- UWAJD (icassp 2008): $\mathbf{V}$ is solution to $\varphi(\mathbf{V}) = \mathbf{I}$,
  $\operatorname{diag}(\mathbf{V}\mathbf{R}_{\mathbf{x}}[0]\mathbf{V}^T) = \mathrm{ones}(d,1)$, where
  $\varphi(\mathbf{V}) = \operatorname{argmin}_{\mathbf{A}} \sum_{\tau=1}^{M} \|\mathbf{V}\mathbf{R}_{\mathbf{x}}[\tau]\mathbf{V}^T - \mathbf{A}\mathbf{D}_{\tau,\mathbf{V}}\mathbf{A}^T\|_{\mathrm{F}}^2$
  $\mathbf{D}_{\tau,\mathbf{V}} = \operatorname{diag}(\operatorname{diag}(\mathbf{V}\mathbf{R}_{\mathbf{x}}[\tau]\mathbf{V}^T))$

AJD algorithms for WASOBI and BGL

- LLAJD (Pham) $\min_{\mathbf{V}} \sum_{\tau=1}^{M} \log \frac{\det \operatorname{diag}(\mathbf{V}\mathbf{R}_{\mathbf{x}}[\tau]\mathbf{V}^T)}{\det(\mathbf{V}\mathbf{R}_{\mathbf{x}}[\tau]\mathbf{V}^T)}$
- ACDC (Yeredor) $\min_{\mathbf{V},\mathbf{D}_\tau} \sum_{\tau=0}^{M-1} \|\mathbf{R}_{\mathbf{x}}[\tau] - \mathbf{V}^{-1}\mathbf{D}_\tau\mathbf{V}^{-T}\|_{\mathrm{F}}^2$
- FAJD (Li, Zhang)
  $\min_{\mathbf{V}} \sum_{\tau=1}^{M} \|\operatorname{off}(\mathbf{V}\mathbf{R}_{\mathbf{x}}[\tau]\mathbf{V}^T)\|_{\mathrm{F}}^2 - \beta \log|\det \mathbf{V}|$
- FFDIAG (Ziehe et al) ... the same as UWAJD ?
- QAJD (Vollgraf, Obermayer) $\min_{\mathbf{V}} \sum_{\tau=1}^{M} \|\operatorname{off}(\mathbf{V}\mathbf{R}_{\mathbf{x}}[\tau]\mathbf{V}^T)\|_{\mathrm{F}}^2$
  subj. to $\operatorname{diag}(\mathbf{V}\mathbf{R}_{\mathbf{x}}[0]\mathbf{V}^T) = \operatorname{ones}(d,1)$
- UWAJD (icassp 2008): $\mathbf{V}$ is solution to $\varphi(\mathbf{V}) = \mathbf{I}$,
  $\operatorname{diag}(\mathbf{V}\mathbf{R}_{\mathbf{x}}[0]\mathbf{V}^T) = \operatorname{ones}(d,1)$, where
  $\varphi(\mathbf{V}) = \operatorname{argmin}_{\mathbf{A}} \sum_{\tau=1}^{M} \|\mathbf{V}\mathbf{R}_{\mathbf{x}}[\tau]\mathbf{V}^T - \mathbf{A}\mathbf{D}_{\tau,\mathbf{V}}\mathbf{A}^T\|_{\mathrm{F}}^2$
  $\mathbf{D}_{\tau,\mathbf{V}} = \operatorname{diag}(\operatorname{diag}(\mathbf{V}\mathbf{R}_{\mathbf{x}}[\tau]\mathbf{V}^T))$

AJD algorithms for WASOBI and BGL

- LLAJD (Pham) $\min_{\mathbf{V}} \sum_{\tau=1}^{M} \log \frac{\det \mathrm{diag}(\mathbf{V}\mathbf{R}_{\mathbf{x}}[\tau]\mathbf{V}^T)}{\det(\mathbf{V}\mathbf{R}_{\mathbf{x}}[\tau]\mathbf{V}^T)}$
- ACDC (Yeredor) $\min_{\mathbf{V},\mathbf{D}_\tau} \sum_{\tau=0}^{M-1} \|\mathbf{R}_{\mathbf{x}}[\tau] - \mathbf{V}^{-1}\mathbf{D}_\tau\mathbf{V}^{-T}\|_{\mathrm{F}}^2$
- FAJD (Li, Zhang)
  $\min_{\mathbf{V}} \sum_{\tau=1}^{M} \|\mathrm{off}(\mathbf{V}\mathbf{R}_{\mathbf{x}}[\tau]\mathbf{V}^T)\|_{\mathrm{F}}^2 - \beta \log|\det \mathbf{V}|$
- FFDIAG (Ziehe et al) ... the same as UWAJD ?
- QAJD (Vollgraf, Obermayer) $\min_{\mathbf{V}} \sum_{\tau=1}^{M} \|\mathrm{off}(\mathbf{V}\mathbf{R}_{\mathbf{x}}[\tau]\mathbf{V}^T)\|_{\mathrm{F}}^2$
  subj. to $\mathrm{diag}(\mathbf{V}\mathbf{R}_{\mathbf{x}}[0]\mathbf{V}^T) = \mathrm{ones}(d,1)$
- UWAJD (icassp 2008): $\mathbf{V}$ is solution to $\varphi(\mathbf{V}) = \mathbf{I}$,
  $\mathrm{diag}(\mathbf{V}\mathbf{R}_{\mathbf{x}}[0]\mathbf{V}^T) = \mathrm{ones}(d,1)$, where
  $\varphi(\mathbf{V}) = \mathrm{argmin}_{\mathbf{A}} \sum_{\tau=1}^{M} \|\mathbf{V}\mathbf{R}_{\mathbf{x}}[\tau]\mathbf{V}^T - \mathbf{A}\mathbf{D}_{\tau,\mathbf{V}}\mathbf{A}^T\|_{\mathrm{F}}^2$
  $\mathbf{D}_{\tau,\mathbf{V}} = \mathrm{diag}(\mathrm{diag}(\mathbf{V}\mathbf{R}_{\mathbf{x}}[\tau]\mathbf{V}^T))$

Difference between QAJD and UWAJD

$$\mathbf{R_x}[1] = \begin{bmatrix} 1 & -0.98 \\ -0.98 & 1 \end{bmatrix}, \mathbf{R_x}[2] = \begin{bmatrix} 35 & 3 \\ 3 & 0.6 \end{bmatrix}, \mathbf{R_x}[3] = \begin{bmatrix} 26 & -0.2 \\ -0.2 & 0.1 \end{bmatrix}$$

QAJD: UW criterion = 2.0

$$\mathbf{R_s}[1] = \begin{bmatrix} 1 & 0.9994 \\ 0.9994 & 1 \end{bmatrix}, \mathbf{R_s}[2] = \begin{bmatrix} 0.44 & 0.00 \\ 0.00 & 0.85 \end{bmatrix}, \mathbf{R_s}[3] = \begin{bmatrix} 0.69 & 0.00 \\ 0.00 & 0.12 \end{bmatrix}$$

UWAJD: UW criterion = 193.7

$$\mathbf{R_s}[1] = \begin{bmatrix} 1 & -0.14 \\ -0.14 & 1 \end{bmatrix}, \mathbf{R_s}[2] = \begin{bmatrix} 1029 & -5.2 \\ -5.2 & 0.3 \end{bmatrix}, \mathbf{R_s}[3] = \begin{bmatrix} 634.5 & 8.4 \\ 8.4 & 0.21 \end{bmatrix}$$

Difference between QAJD and UWAJD

$$\mathbf{R_x}[1] = \begin{bmatrix} 1 & -0.98 \\ -0.98 & 1 \end{bmatrix}, \mathbf{R_x}[2] = \begin{bmatrix} 35 & 3 \\ 3 & 0.6 \end{bmatrix}, \mathbf{R_x}[3] = \begin{bmatrix} 26 & -0.2 \\ -0.2 & 0.1 \end{bmatrix}$$

QAJD: UW criterion = 2.0

$$\mathbf{R_s}[1] = \begin{bmatrix} 1 & 0.9994 \\ 0.9994 & 1 \end{bmatrix}, \mathbf{R_s}[2] = \begin{bmatrix} 0.44 & 0.00 \\ 0.00 & 0.85 \end{bmatrix}, \mathbf{R_s}[3] = \begin{bmatrix} 0.69 & 0.00 \\ 0.00 & 0.12 \end{bmatrix}$$

UWAJD: UW criterion = 193.7

$$\mathbf{R_s}[1] = \begin{bmatrix} 1 & -0.14 \\ -0.14 & 1 \end{bmatrix}, \mathbf{R_s}[2] = \begin{bmatrix} 1029 & -5.2 \\ -5.2 & 0.3 \end{bmatrix}, \mathbf{R_s}[3] = \begin{bmatrix} 634.5 & 8.4 \\ 8.4 & 0.21 \end{bmatrix}$$

Difference between QAJD and UWAJD

$$\mathbf{R_x}[1] = \begin{bmatrix} 1 & -0.98 \\ -0.98 & 1 \end{bmatrix}, \mathbf{R_x}[2] = \begin{bmatrix} 35 & 3 \\ 3 & 0.6 \end{bmatrix}, \mathbf{R_x}[3] = \begin{bmatrix} 26 & -0.2 \\ -0.2 & 0.1 \end{bmatrix}$$

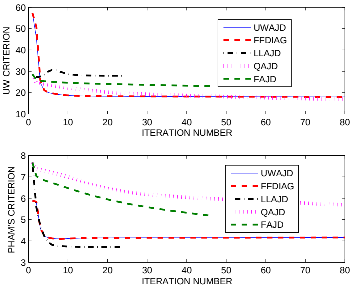
QAJD: UW criterion = 2.0

$$\mathbf{R_s}[1] = \begin{bmatrix} 1 & 0.9994 \\ 0.9994 & 1 \end{bmatrix}, \mathbf{R_s}[2] = \begin{bmatrix} 0.44 & 0.00 \\ 0.00 & 0.85 \end{bmatrix}, \mathbf{R_s}[3] = \begin{bmatrix} 0.69 & 0.00 \\ 0.00 & 0.12 \end{bmatrix}$$

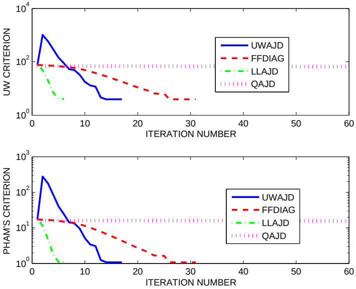
UWAJD: UW criterion = 193.7

$$\mathbf{R_s}[1] = \begin{bmatrix} 1 & -0.14 \\ -0.14 & 1 \end{bmatrix}, \mathbf{R_s}[2] = \begin{bmatrix} 1029 & -5.2 \\ -5.2 & 0.3 \end{bmatrix}, \mathbf{R_s}[3] = \begin{bmatrix} 634.5 & 8.4 \\ 8.4 & 0.21 \end{bmatrix}$$

Learning curves of the AJD techniques



(a) $d = 10, M = 10, \sigma = 0.2$



(b) $d = 100, M = 10, \sigma = 0.01.$

Running times of the AJD techniques

Running times of AJD of $M = 10$ matrices of the size 100×100

Algorithm	positive definite	indefinite
LLAJD	32 s (15 it.)	∞
FFDIAG	8.7 s (30 it)	8.7 s (30 it)
QAJD	126 s (100 it.)	36.4 s
UWAJD	2.9 s (15 it.)	2.9 s (15 it.)

How UWAJD/WAJD works

$\mathbf{V}^{[0]} = ([\mathbf{R_x}[0])^{-1/2}$
For $i = 0, 1, \ldots$ (5 iterations usually suffices) **do**:

- $\mathbf{A}^{[i]} = \mathrm{argmin}_{\mathbf{A}} \sum_{\tau=1}^{M} \|\mathbf{V}^{[i]}\mathbf{R_x}[\tau]\mathbf{V}^{[i]T} - \mathbf{A}\mathbf{D}_{\tau,\mathbf{V}}\mathbf{A}^T\|_{\mathrm{F}}^2$ (Do one step of the Gauss iteration method started from $\mathbf{A} = \mathbf{I}$)
- $\mathbf{V}^{[i+1]} = (\mathbf{A}^{[i]})^{-1}\mathbf{V}^{[i]}$

End

WAJD is a generalization of UWAJD in terms of minimizing a quadratic criterion with an arbitrary positive definite weight matrix in place of the sum of square Frobenius norms.

How UWAJD/WAJD works

$\mathbf{V}^{[0]} = ([\mathbf{R}_\mathbf{x}[0]])^{-1/2}$

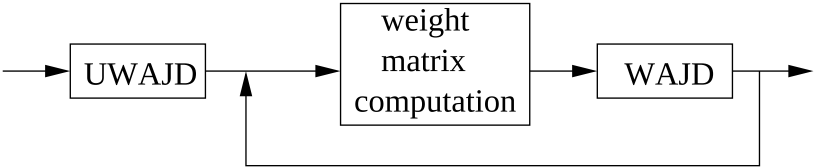
For $i = 0, 1, \ldots$ (5 iterations usually suffices) **do**:

- $\mathbf{A}^{[i]} = \mathrm{argmin}_\mathbf{A} \sum_{\tau=1}^{M} \|\mathbf{V}^{[i]}\mathbf{R}_\mathbf{x}[\tau]\mathbf{V}^{[i]T} - \mathbf{A}\mathbf{D}_{\tau,\mathbf{V}}\mathbf{A}^T\|_\mathrm{F}^2$ (Do one step of the Gauss iteration method started from $\mathbf{A} = \mathbf{I}$)
- $\mathbf{V}^{[i+1]} = (\mathbf{A}^{[i]})^{-1}\mathbf{V}^{[i]}$

End

WAJD is a generalization of UWAJD in terms of minimizing a quadratic criterion with an arbitrary positive definite weight matrix in place of the sum of square Frobenius norms.

General scheme of WASOBI



WASOBI, WASOBI-like implementation of BGL, block WASOBI

Features of WASOBI

- If all source signals are Gaussian AR processes of order $M-1$ the asymptotic ISR matrix is equal to the corresponding CRLB:

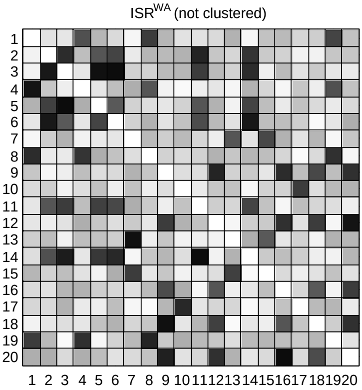
$$\mathbf{ISR}^{WA}_{k\ell} = \mathbf{CRLB}_{k\ell} = \frac{1}{N}\frac{\phi_{k\ell}}{1-\phi_{k\ell}\phi_{\ell k}}\frac{\sigma_k^2 R_\ell[0]}{\sigma_\ell^2 R_k[0]}$$

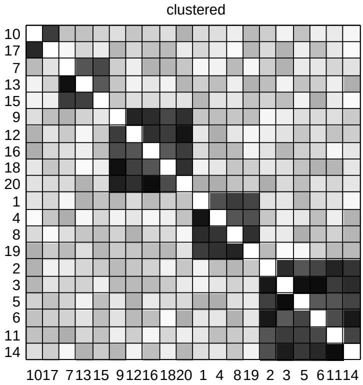
where σ_k^2 is the variance of the innovation sequence of the $k-$th source,

$$\phi_{k\ell} = \frac{1}{\sigma_k^2}\sum_{i,j=0}^{M-1} a_{i\ell}a_{j\ell}R_k[i-j]$$

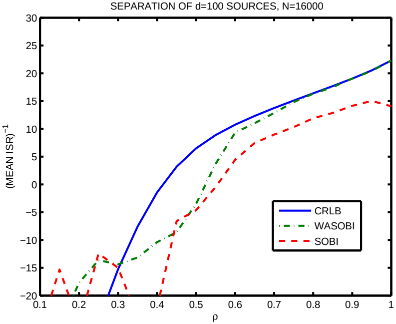
and $\{a_{i\ell}\}_{i=0}^{M-1}$ are AR coefficients of the $\ell-$th source with $a_{0\ell}=1$ for $k,\ell = 1,\ldots,d$.

Clustering of WASOBI ISR matrix





Example of performance of WASOBI

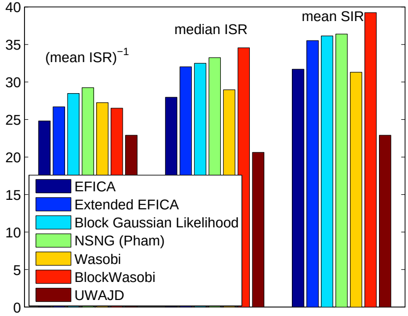


Quality of separation of 100 Gaussian AR(10) sources with poles at $p_k^{(i)} e^{\pm j\pi k/6}$, $k = 1, \ldots, 5$, where $p_k^{(i)} \in \{0.6\rho, 0.85\rho, 0.95\rho\}$, versus parameter ρ.

Combinations of the techniques

- EFICA & WASOBI => COMBI, Multi-COMBI
- Joint diagonalization of cumulant slices and lagged covariance matrices (ThinICA, JADE_{TD})
- Block EFICA - Extended EFICA
- Block WASOBI
- Pham's algorithm combining nonstationarity and nonGaussianity

Separation of a linear mixture of speech signals



10 speech signals

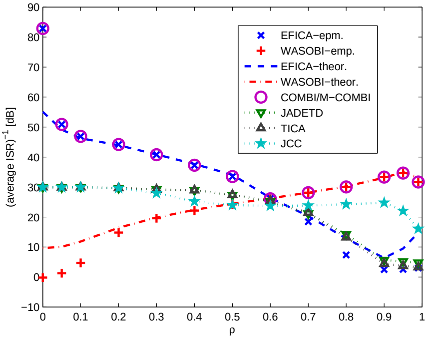
$N = 5000$

40 blocks

AR(10) for WASOBI, AR(2) for block WASOBI

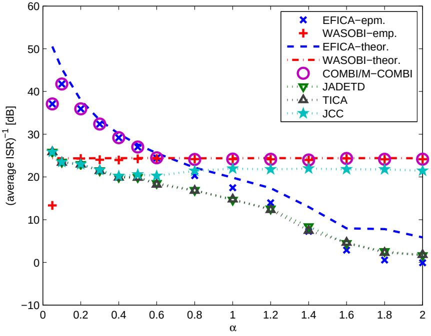
1000 trials

COMBI: Validity of the ISR expressions in model mismatch



Separation of five AR signals obtained by passing binary (BPSK) i.i.d. sequences of length $N = 1000$ through all-pole filters with autoregression coefficients $[1, \rho]$, $[1, 0, \rho]$, $[1, 0, 0, \rho]$, $[1, 0, 0, 0, \rho]$, and $[1, 0, 0, 0, 0, \rho]$, respectively.

Validity of the ISR expressions in a model mismatch



Results for sources obtained by passing GG(α)-distributed sequences through the filters with coefficients $[1, \rho]$, $[1, 0, \rho]$, $[1, 0, 0, \rho]$, $[1, 0, 0, 0, \rho]$, and $[1, 0, 0, 0, 0, \rho]$ for $\rho = 0.5$ versus varying α.

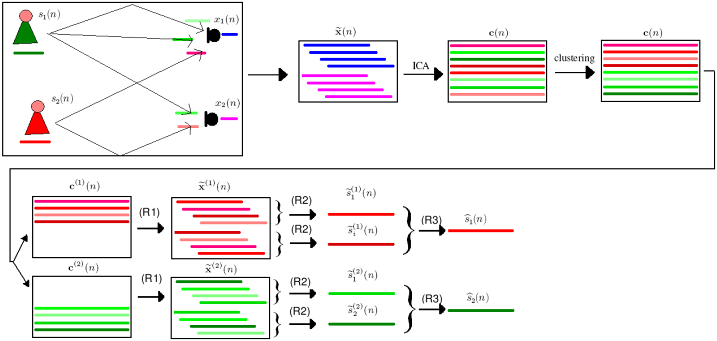
Time-domain separation of a convolutive mixture

Idea: Independent components will contain inovative sequences of the sources and their time-shifted copies.

Method:

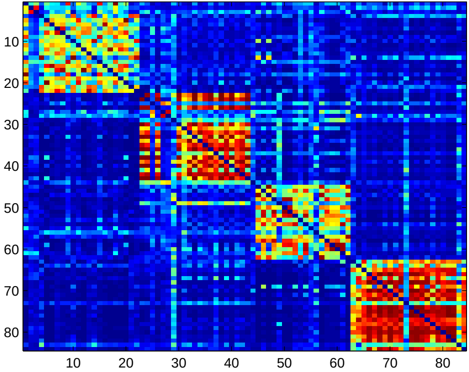
1. Application of an ICA technique to time-shifted copies of the signals
2. Computation of matrix of distances between the IC's
3. Clustering of the independent components
4. Reconstruction of the separated sources

Time-domain separation of a convolutive mixture



Time-domain separation of a convolutive mixture

Matrix of “distances” between the “independent” components



Four speakers
21 time lags

Conclusions ? Too many !

Thank you for your attention !