Codebook-based, Single Channel Blind Source Separation of Audio Signals

Guy Rapaport

Electrical Engineering department

Technion – Israel Institute of Technology

Supervised by Prof. Israel Cohen

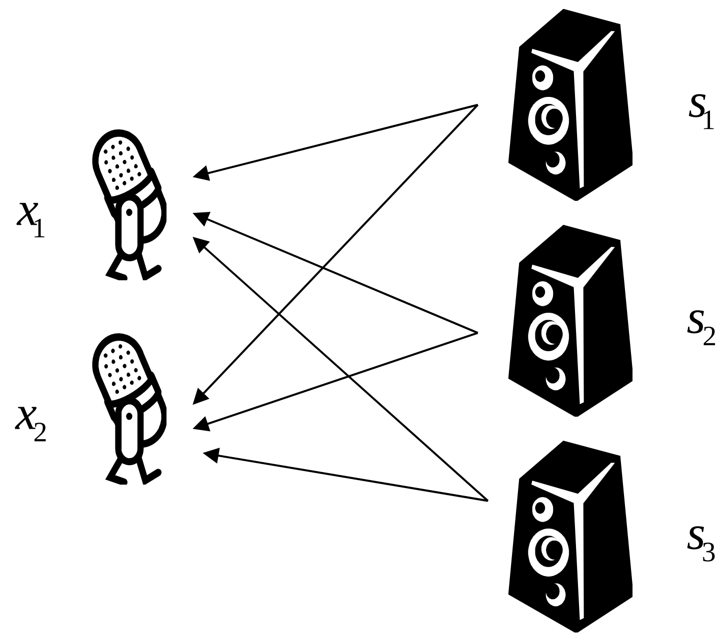
Outline

- Introduction
  - Blind Source Separation (BSS)
  - Single Channel BSS
- Codebook-based Single Channel BSS
  - NMF/GMM/AR
- Separation Cost function
  - Frequency Dependent Separation
  - Distant Power Spectral Densities (PSDs)
- Experimental Study
- Conclusions

Blind Source Separation (1)

- Problem definition -

  Separating N sources from M observations



Blind Source Separation (2)

- The separation problem depends on –
  - Mixing model:
    - Instantaneous (linear mixtures)
    - Un-echoic (introducing delays)
    - Echoic (reverberant environment)

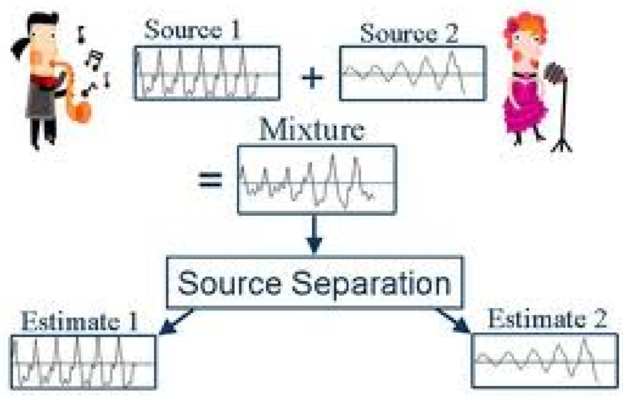
Blind Source Separation (3)

- The separation problem depends on -
  - Number of sources (N) and observations (M)
    - Over-determined ($M \geq N$) scenario
      Many separation methods exist
      - **[O'Grady at el., 2005]**
        A survey of methods for source separation

    - Under-determined ($M < N$) scenario
      Prior knowledge on the sources is needed

Single Channel BSS (1)

- The most extreme case of the under-determined separation problem
- Objective:
  separating two (or more) sources from their mixture
- Assumptions:
  - Instantaneous mixing model
  - Main focus on audio signals



Single Channel BSS (2)

- Single channel BSS requires priors for successful separation
- These priors may originate from -
  - Computational Auditory Scene Analysis (CASA)
  - Independent Component Analysis
  - Predefined Codebook of the sources

Single Channel BSS (3)

- CASA-based separation methods –
  - Mimic psycho-acoustics characteristics of the Human Auditory System
  - Use perceptual cues as heuristics in the source separation scheme
- ICA-based separation methods –
  - Adjusting the ICA-based solution from the over-determined realm into the single channel scenario

Single Channel BSS (4)

- CASA-based separation methods –
  - Examples:
    - **[Roweis, 2001]**
      Assumes only one signal is dominant per T-F bin
    - **[Duan, 2004]**
      Music separation according to harmonic structure
    - **[Bach & Jordan, 2003]**
      Source separation via spectral clustering. The clustering cost function is using a CASA-driven features

Single Channel BSS (5)

- ICA-based separation methods –
  - Examples:
    - **[Jang & Lee, 2003]**
      Describing each source, in the time domain, as a mixture of statistically independent components
    - **[Beierholm et al., 2004]**
      Similar to Jang and Lee, only in the DCT domain
    - **[Mijovic et al., 2010]**
      ICA-based separation method in the wavelet domain or following a dedicated data-driven transform

Codebook-based Single Channel BSS (1)

- Prior –
  A codebook (CB) is used for representing each source
  - The CB describes the source according to a selected representation model
  - Requires an offline learning stage
    - Train signals - similar to the source in the mixture
    - Clustering the train observations into CBs

Codebook-based Single Channel BSS (2)

- In the context of audio signals
  - Time domain representation –

$$x[n] \approx s_1[n] + s_1[n]$$

  - In the STFT domain -

$$X(f,t) \approx S_1(f,t) + S_2(f,t)$$

  - Separation is achieved by estimating the sources' power spectral densities (PSDs) that will best match the mixture's PSD -

$$P_x(f,t) \approx P_1(f,t) + P_2(f,t)$$

Codebook-based Single Channel BSS (3)

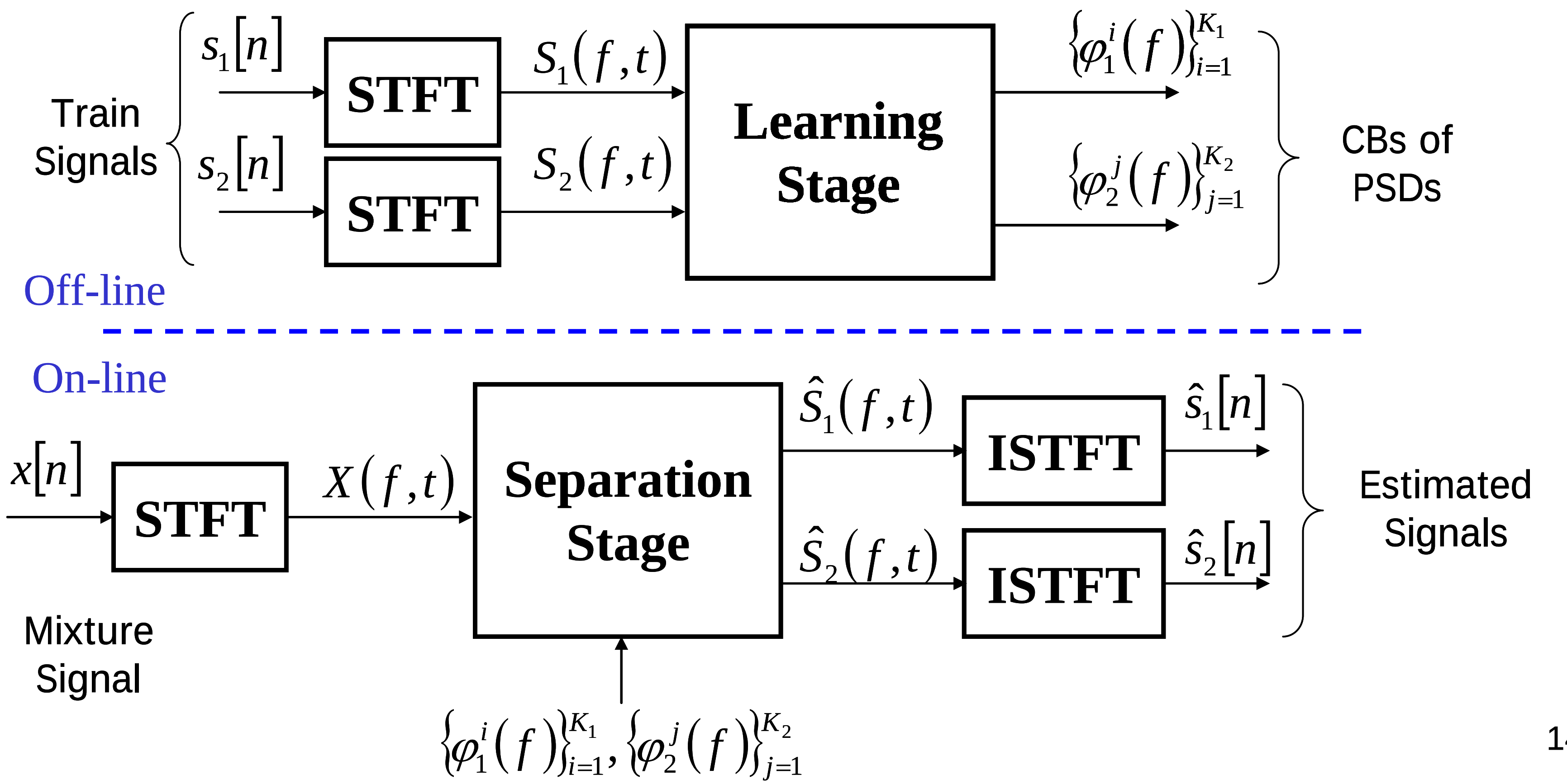
- CB of PSDs:
  - The Source's PSD are represented by a codebook of PSDs
    - The gain factors are non-negative

$$\begin{cases} P_1(f,t) = \sum_{i=1}^{K_1} a_1^i(t) \cdot \varphi_1^i(f) \\ P_2(f,t) = \sum_{j=1}^{K_2} a_2^j(t) \cdot \varphi_2^j(f) \end{cases}$$

  - Aim to separate a quasi-stationary mixture with a time-varying combination of stationary spectral shapes

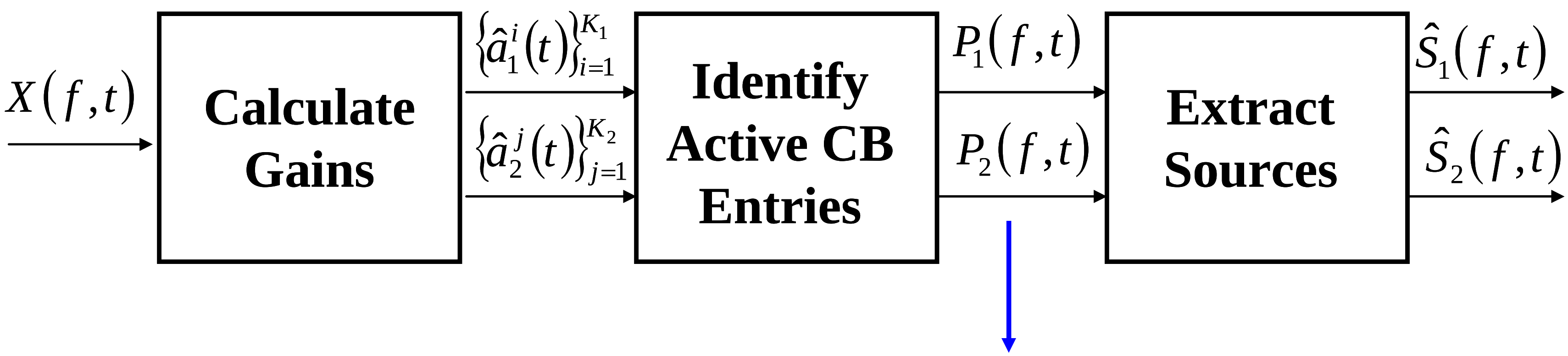
Codebook-based Single Channel BSS (4)

- Separation stages – overview:



Codebook-based Single Channel BSS (5)

- Separation stage:



$$\begin{cases} P_1(f,t) = \sum_{i=1}^{K_1} \hat{a}_1^i(t) \cdot \varphi_1^i(f) \\ P_2(f,t) = \sum_{j=1}^{K_2} \hat{a}_2^j(t) \cdot \varphi_2^j(f) \end{cases}$$

Codebook-based Single Channel BSS (6)

- Several types of CB-based separation algorithms -
  - Non-negative Matrix Factorization (NMF)
  - Gaussian Mixture Model (GMM)
  - Auto Regressive (AR) model

- All are eventually evolving to CBs of PSDs

NMF-based Separation (1)

[Lee & Seung, 2001]

- Non-negative Matrix Factorization
  - Efficient decomposition method

$$P = B \cdot G$$

    - P,G,B – non-negative matrices
  - Two cost functions
    - Frobenious norm – $\|P - BG\|_F^2 = \sum_{i,j} \left(P_{i,j} - (BG)_{i,j}\right)^2$
    - KL Divergence - $\sum_{i,j} \left(P_{i,j} \cdot \log \frac{P_{i,j}}{(BG)_{i,j}} - P_{i,j} + (BG)_{i.j} \right)$

NMF-based Separation (2)

- Non-negative Matrix Factorization (cont.)
  - Using a multiplicative update rule
    - $\otimes$ - represent element-wise multiplication

Frobenious norm:

$$\begin{cases} B = B \otimes \left(\dfrac{PG^T}{BGG^T} \right) \\ G = G \otimes \left(\dfrac{B^T P}{B^T BG} \right) \end{cases}$$

KL divergence:

$$\begin{cases} B = B \otimes \left(\dfrac{\dfrac{P}{BG} G^T}{1 \cdot G^T} \right) \\ G = G \otimes \left(\dfrac{B^T \dfrac{P}{BG}}{B^T \cdot 1} \right) \end{cases}$$

NMF-based Separation (3)

- Separation algorithm

$$P_x(f,t) \approx P_1(f,t) + P_2(f,t)$$

- Where –

$$\begin{cases} P_1(f,t) = \sum_{i=1}^{K_1} a_1^i(t) \cdot \varphi_1^i(f) \\ P_2(f,t) = \sum_{j=1}^{K_2} a_2^j(t) \cdot \varphi_2^j(f) \end{cases}$$

- Matrix Notation -

Basis (CB) Matrix

Gain Matrix

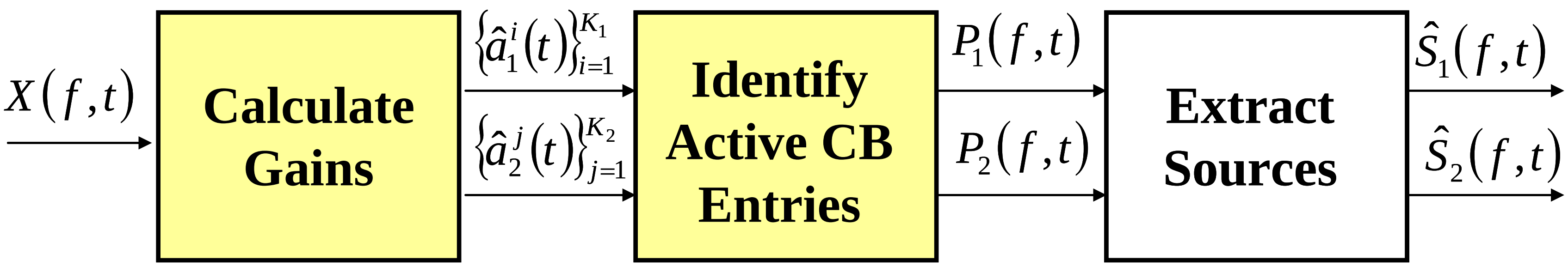
Observed PSD matrix

$$P_x \approx P_1 + P_2 = B_1 \cdot G_1 + B_2 \cdot G_2 = \underbrace{\begin{bmatrix} B_1 & B_2 \end{bmatrix}}_{B} \cdot \underbrace{\begin{bmatrix} G_1 \\ G_2 \end{bmatrix}}_{G}$$

NMF-based Separation (4)

- Separation algorithm (cont.)
  - Basis matrix:
    columns contains the CB entries $\left\{\varphi_1^i(f)\right\}_{i=1}^{K_1}, \left\{\varphi_2^j(f)\right\}_{j=1}^{K_2}$
  - Gain matrix:
    rows contains the time-varying gains $\left\{a_1^i(t)\right\}_{i=1}^{K_1}, \left\{a_2^j(t)\right\}_{j=1}^{K_2}$
- Learning stage **[Wang & Plumbley, 2006]**
  - Run NMF on the training data to extract B_1, B_2 as the PSD CB of each source

NMF-based Separation Algorithmic Flow (1)



- Run NMF
  - Only update the gain matrix
- Estimate Sources' PSD

$$\begin{cases} P_1(f,t) = (B_1 G_1)_{f,t} \\ P_2(f,t) = (B_2 G_2)_{f,t} \end{cases}$$

NMF-based Separation Algorithmic Flow (2)

- Wiener filtering

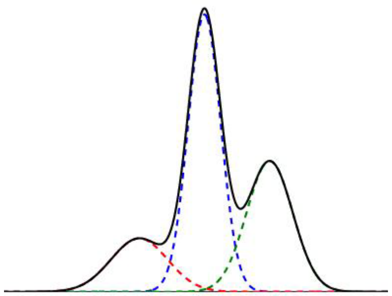
$$\hat{S}_1(f,t) = \frac{P_1(f,t)}{P_1(f,t) + P_2(f,t)} \cdot X(f,t)$$

GMM-based Separation (1)

[Benaroya et al., 2006]

- The sources are statistically independent
- Each source is represented by a GMM

$$p(s) = \sum_{i=1}^{K} p(s \mid \theta_i) \cdot \Pr(\theta_i)$$



  - $s \mid \theta_i \sim N(0, \Sigma_i)$
  - $\Pr(\theta_i)$ - Prior probability of θ_i
  - EM is used for training

GMM-based Separation (2)

- Task:
  Separating the mixture – $x[n] = s_1[n] + s_2[n]$
- Simple case
  - The active Gaussian components ,(θ_1^i, θ_2^j) , are known
    - Both sources are Gaussian
  - Wiener filtering can be used for separation

$$s_1 = \Sigma_1^i \cdot \left(\Sigma_1^i + \Sigma_2^j\right)^{-1} \cdot x$$

GMM-based Separation (3)

- Simple case (cont.)
  - STFT domain solution:
    - Assume: the signals are quasi-stationary and approximately circular

$$\Sigma_1^i \to \varphi_1^i(f),\ \Sigma_2^j \to \varphi_2^j(f)$$

$$\Downarrow$$

$$S_1(f,t) = \frac{\varphi_1^i(f)}{\varphi_1^i(f) + \varphi_2^j(f)} \cdot X(f,t)$$

GMM-based Separation (4)

- General case
  - The active components are **unknown**
    - Bayesian formalism

$$p\left(\theta_1^i,\theta_2^j \mid x\right) \propto p\left(x \mid \theta_1^i,\theta_2^j\right) \cdot \Pr\left(\theta_1^i\right) \cdot \Pr\left(\theta_2^j\right)$$

    - MAP criterion

$$\left(i^*, j^*\right) = \underset{i,j}{\operatorname{argmax}} \left\{ p\left(\theta_1^i,\theta_2^j \mid x\right) \right\}$$

    - We are back in the simple case scenario...

GMM-based Separation (5)

- General case (cont.)
  - STFT domain interpretation
    - Covariance matrices turns to CB of PSDs

$$\left\{\Sigma_1^i\right\}_{i=1}^{K_1} \rightarrow \left\{\varphi_1^i(f)\right\}_{i=1}^{K_1}, \left\{\Sigma_2^j\right\}_{j=1}^{K_2} \rightarrow \left\{\varphi_2^j(f)\right\}_{j=1}^{K_2}$$

    - MAP criterion

$$\left(i^*(t), j^*(t)\right) = \underset{i,j}{\operatorname{argmax}} \left\{ \overbrace{p\left(X(f,t) \mid \theta_1^i, \theta_2^j\right)}^{Gaussian\ distribution} \cdot \Pr\left(\theta_1^i\right) \cdot \Pr\left(\theta_2^j\right) \right\}$$

    - No attention for gain estimation

GMM-based Separation (6)

- GSMM approach
  - Gaussian **Scaled** Mixture Model
  - Introducing a gain for each Gaussian component
    - If the gains $\left\{a^i\right\}_{i=1}^{K} \geq 0$ are known: GSMM $\rightarrow$ GMM
      - With covariance matrices - $\left\{a^i \cdot \Sigma^i\right\}_{i=1}^{K}$
  - GSMM requires an additional gain estimation stage (prior to the pair selection)
    - ML criterion

$$\left(\hat{a}_1^i(t), \hat{a}_2^j(t)\right) = \underset{\left(a_1^i, a_2^j\right) \geq 0}{\operatorname{argmax}} \left\{p\left(X(f,t) \mid \theta_1^i, \theta_2^j, a_1^i, a_2^j\right)\right\}$$

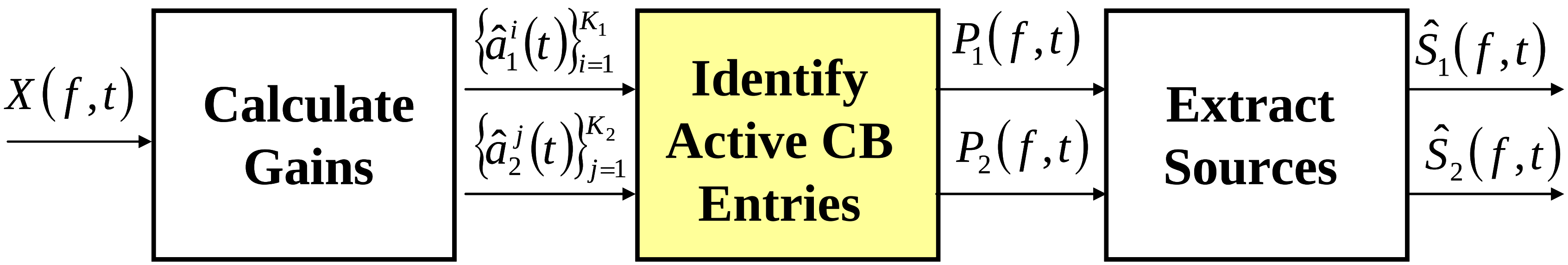
GMM-based Separation Algorithmic Flow (1)

- ML criterion

$$\left(\hat{a}_1^i(t), \hat{a}_2^j(t)\right) = \underset{\left(a_1^i, a_2^j\right) \geq 0}{\operatorname{argmax}} \left\{ p\left(X(f,t) \mid \theta_1^i, \theta_2^j, a_1^i, a_2^j\right) \right\}$$

  - Solved via multiplicative update rule (NMF-like)

GMM-based Separation Algorithmic Flow (2)



- Choosing the optimal pair

$$\left(i^*(t), j^*(t)\right) = \underset{i,j}{\operatorname{argmax}} \left\{ p\left(X(f,t) \mid \theta_1^i, \theta_2^j, \hat{a}_1^i(t), \hat{a}_2^j(t)\right) \cdot \Pr\left(\theta_1^i\right) \cdot \Pr\left(\theta_2^j\right) \right\}$$

GMM-based Separation Algorithmic Flow (3)

- Wiener filtering

$$\hat{S}_1(f,t)=\frac{\hat{a}_1^{i^*}(t)\cdot\varphi_1^{i^*}(f)}{\underbrace{\hat{a}_1^{i^*}(t)\cdot\varphi_1^{i^*}(f)}_{P_1(f,t)}+\underbrace{\hat{a}_2^{j^*}(t)\cdot\varphi_2^{j^*}(f)}_{P_2(f,t)}}\cdot X(f,t)$$

AR-based Separation (1)

[Srinivasan et al., 2006]

- Originated from speech enhancement methods
  - Traditionally – noise is slowly changing in comparison to the speech signal
  - What if the undesired signal is changing rapidly?
    - A different prior is needed...
- Proposition:
  - At each time frame, model the sources as AR processes
  - A CB of AR processes is available for each source

AR-based Separation (2)

- AR process
  - Used for representing speech spectral shapes
  - Definition:

    AR of order P – $$s[n]=\sum_{i=1}^{P}\alpha_i \cdot s[n-i]+u[n]$$

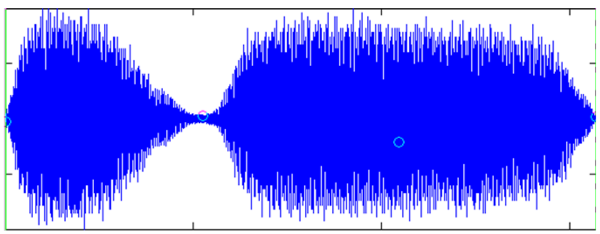
    - $\theta=\{\alpha_i\}_{i=1}^{P}$ - Linear Prediction Coefficients (LPC)
    - $u[n]\sim N(0,\sigma^2)$
    - σ^2 - Excitation variance

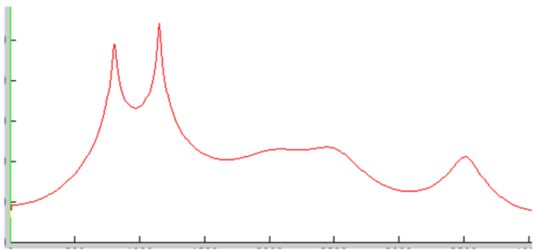
AR-based Separation (3)

- AR process (cont.)
  - Spectral envelope

$$P(f)=\frac{\sigma^2}{|A(f)|^2}\ ,\ A(f)=1+\sum_{i=n}^{P}\alpha_n\cdot e^{-2\pi j\cdot fn}$$

- $\varphi(f)=\frac{1}{|A(f)|^2}$ - Spectral shape
- σ^2 - Gain factor (amplitude)





AR-based Separation (4)

- Task:
  Separating the mixture – $x[n] = s_1[n] + s_2[n]$
- Solution:
  - CB of AR processes for each source –

$$\left\{\theta_1^i\right\}_{i=1}^{K_1},\ \left\{\theta_2^j\right\}_{j=1}^{K_2} \quad \to \quad \left\{\varphi_1^i(f)\right\}_{i=1}^{K_1},\ \left\{\varphi_2^j(f)\right\}_{j=1}^{K_2}$$

  - Learning stage
    - Clustering LPCs via Max Lloyd algorithm

AR-based Separation (5)

- Solution (cont.)
  - ML criterion

$$\left(i^*, j^*\right) = \underset{(i,j)}{\operatorname{argmax}} \left\{ \max_{\left(\sigma_1^2, \sigma_2^2\right) \geq 0} \left\{ p\left(x \mid \theta_1^i, \theta_2^j, \sigma_1^2, \sigma_2^2\right) \right\} \right\}$$

Find CB Representatives

Estimating gains

AR-based Separation (5)

- Solution (cont.)
  - STFT representation

$$\left(i^*, j^*\right) = \underset{(i,j)}{\operatorname{argmin}} \left\{ \min_{\left(\sigma_1^2, \sigma_2^2\right) \geq 0} \left\{ D_{IS}\left(P_x(f,t), P_1(f,t) + P_2(f,t)\right) \right\} \right\}$$

  - Itakura-Saito distortion measure (D_{IS})
    - Widely used for measuring distance between PSDs

$$D_{IS}\left(P_x, P_y\right) = \frac{1}{F} \sum_{f=0}^{F-1} \left[\frac{P_x(f)}{P_y(f)} - \log\left(\frac{P_x(f)}{P_y(f)} \right) - 1 \right]$$

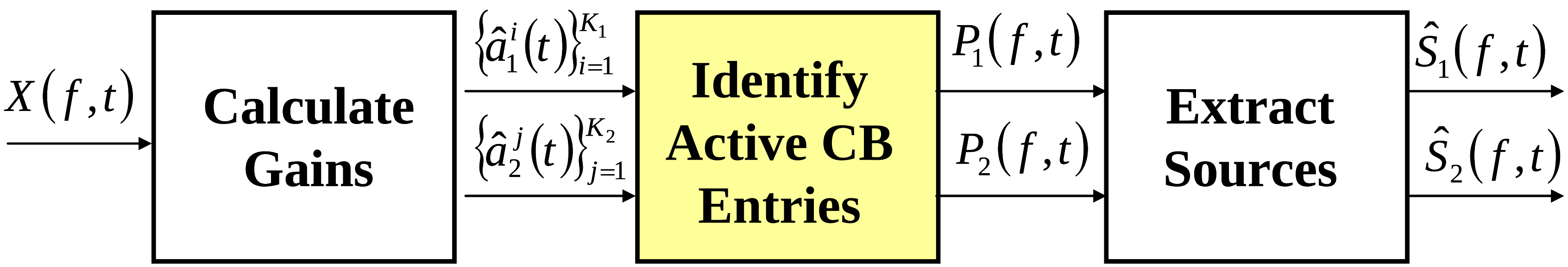
AR-based Separation Algorithmic Flow (1)

- ML criterion –

$$\left(\hat{a}_1^i(t),\hat{a}_2^j(t)\right)=\underset{\left(a_1^i,a_2^j\right)\geq 0}{\operatorname{argmin}}\left\{D_{IS}\left(P_x(f,t),P_1(f,t)+P_2(f,t)\right)\right\}$$

  - Solved by linearization

AR-based Separation Algorithmic Flow (2)



- Choosing the optimal pair

$$\left(i^*, j^*\right) = \underset{(i,j)}{\operatorname{argmin}} \left\{ D_{IS}\left(P_x(f,t), P_1(f,t) + P_2(f,t)\right) \right\}$$

  - Where
$$\begin{cases} P_1(f,t) = \hat{a}_1^i(t) \cdot \varphi_1^i(f) \\ P_2(f,t) = \hat{a}_2^j(t) \cdot \varphi_2^j(f) \end{cases}$$

AR-based Separation Algorithmic Flow (3)

- Wiener filtering

$$\hat{S}_1(f,t)=\frac{\hat{a}_1^{i^*}(t)\cdot\varphi_1^{i^*}(f)}{\hat{a}_1^{i^*}(t)\cdot\varphi_1^{i^*}(f)+\hat{a}_2^{j^*}(t)\cdot\varphi_2^{j^*}(f)}\cdot X(f,t)$$

AR-based BSS Extensions (1)

- Several suggested alterations TODO!!!
  - MMSE approach **[Benaroya et al., 2006]**
    - Similar to the MAP estimator (but more complicated)
    - All pairs are participating in the separation stage
      - Using a weighted combination of Wiener filters

$$S_1(f,t)=\sum_{i,j} p\left(\theta_1^i,\theta_2^j \mid X(f,t)\right)\cdot\frac{\hat{a}_1^i(t)\cdot\varphi_1^i(f)}{\hat{a}_1^i(t)\cdot\varphi_1^i(f)+\hat{a}_2^j(t)\cdot\varphi_2^j(f)}\cdot X(f,t)$$

    - Separation quality – ~identical to MAP

AR-based BSS Extensions (2)

- **[Benaroya et al., 2003]**

  Using Hidden Markov Model (HMM) in order to describe time-correlation between adjacent frames

- **[Ozerov et al., 2005,2007]**

  On-line update of the sources' CBs using EM for voice/music separation (Requires VAD)

- **[Abramson & Cohen, 2008]**

  Introducing a classification and estimation approach on-top the GMM-based separation method

- **[Emiya et al., 2009]**

  Learning a CB of the mixture instead of the sources

- **[Litvin & Cohen, 2010]**

  Working in the Bark-scale wavelet domain instead of STFT

CB Separation Methods - Observations (1)

- Separation in the STFT domain
  - CBs of PSDs
- CB entries selection
  - GSMM/AR methods seeks for the optimal pair
  - NMF allows all entries to be active
- Cost function
  - GSMM/AR both use the Itakura-Saito distortion measure (SHOW HOW)

$$\underset{(i,j)}{\operatorname{argmin}}\left\{D_{IS}\left(P_x(f,t),P_1(f,t)+P_2(f,t)\right)\right\}=\underset{(i,j)}{\operatorname{argmax}}\left\{p\left(x\mid\theta_1^i,\theta_2^j,a_1^i,a_2^j\right)\right\}$$

CB Separation Methods - Observations (2)

- Current separation results –
  not good enough!

- Improvements?
  - Diving into the separation cost function

Separation Cost Function (1)

- AR/GMM cost function

$$D_{IS}\left(P_x(f,t), \underbrace{P_{1+2}(f,t)}_{P_1(f,t)+P_2(f,t)} \right) = \frac{1}{F}\sum_{f=0}^{F-1}\left[\frac{P_x(f)}{P_{1+2}(f)} - \log\left(\frac{P_x(f)}{P_{1+2}(f)} \right) - 1 \right]$$

- Observation
  - Each frequency bin is treated <u>identically</u>

Separation Cost Function (2)

- But –
  - Frequency bins with sufficient energy should be more "important" than noisy bins
  - Wiener filtering – accurate PSD estimation is not important where $\left|X\left(f,t\right)\right| \approx 0$
  - What if a signal is band limited?
  - CASA-motivated frequency differentiation

Frequency Dependent Cost Function (1)

- Generalizing the cost function

$$\tilde{D}_{IS}\left(P_x(f,t),P_{1+2}(f,t)\right)=\frac{1}{F}\sum_{f=0}^{F-1}\lambda_f\left[\frac{P_x(f)}{P_{1+2}(f)}-\log\left(\frac{P_x(f)}{P_{1+2}(f)}\right)-1\right]$$

- $\{\lambda_f\}_{f=0}^{F-1}$ - frequency weights
  - If $\lambda_f=1\,,\forall f$ -
    we are back to the regular IS distortion measure

Frequency Dependent Cost Function (2)

- Probability function interpretation
  - Following the connection

$$\underset{(i,j)}{\operatorname{argmin}}\left\{D_{IS}\left(P_x(f,t),P_{1+2}(f,t)\right)\right\}=\underset{(i,j)}{\operatorname{argmax}}\left\{p\left(X(f,t)\mid\theta_1^i,\theta_2^j,a_1^i,a_2^j\right)\right\}$$

  - Where -

$$p\left(X(f,t)\mid\theta_1^i,\theta_2^j,a_1^i,a_2^j\right)\propto\prod_{f=0}^{F-1}\left\{\left[P_{1+2}(f,t)\right]^{-\frac{1}{2}}\cdot\exp\left(-\frac{P_x(f,t)}{2P_{1+2}(f,t)}\right)\right\}$$

  - Turns to -

$$\tilde{p}\left(X(f,t)\mid\theta_1^i,\theta_2^j,a_1^i,a_2^j\right)\propto\prod_{f=0}^{F-1}\left\{\left[P_{1+2}(f,t)\right]^{-\frac{1}{2}}\cdot\exp\left(-\frac{P_x(f,t)}{2P_{1+2}(f,t)}\right)\right\}^{\lambda_f}$$

Frequency Dependent Cost Function (3)

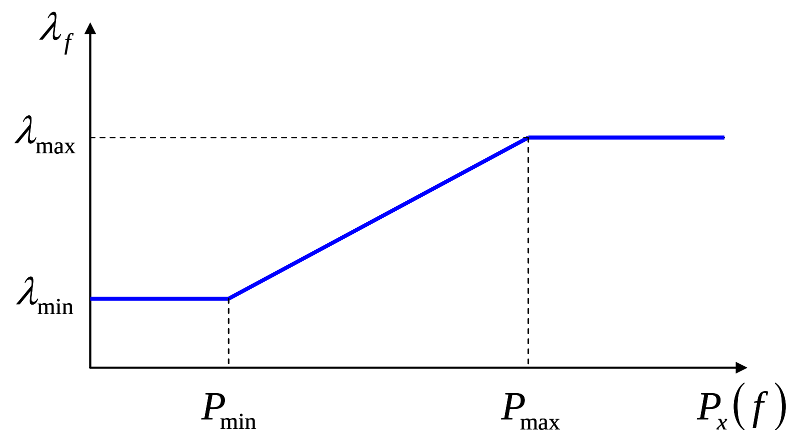
- Probability function interpretation (cont.)

$$\tilde{p}\left(X(f,t) \mid \theta_1^i, \theta_2^j, a_1^i, a_2^j\right) \propto \prod_{f=0}^{F-1} \left\{ \left[P_{1+2}(f,t)\right]^{-\frac{1}{2}} \cdot \exp\left(-\frac{P_x(f,t)}{2P_{1+2}(f,t)} \right) \right\}^{\lambda_f}$$

  - Each Gaussian component is weighted according to $\left\{\lambda_f\right\}_{f=0}^{F-1}$

Frequency Dependent Cost Function (4)

- How to choose $\{\lambda_f\}_{f=0}^{F-1}$?
  - According to the observed PSD of the mixture
    - Example



  - According to the learning stage
    - Identifying the spectral content of each source

Frequency Dependent Separation Algorithm (1)

Following the cost function alteration

- New separation algorithm evolves
  - Based on GSMM-MAP separation method
  - Can also be applied to AR-based separation methods
- Estimates the sources' PSDs while giving different attention to each frequency bin

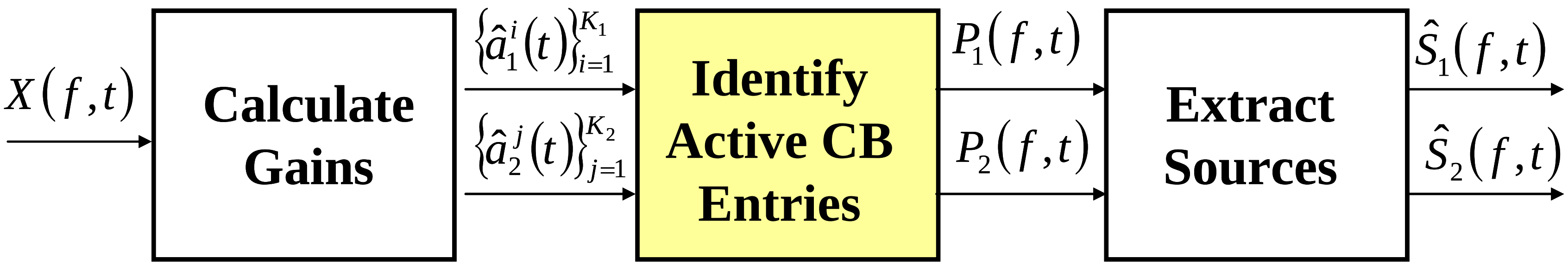
Frequency Dependent Separation Algorithm (2)

- ML criterion

$$\left(\hat{a}_1^i(t), \hat{a}_2^j(t)\right) = \underset{\left(a_1^i, a_2^j\right) \geq 0}{\operatorname{argmax}} \left\{ \tilde{p}\left(X(f,t) \mid \theta_1^i, \theta_2^j, a_1^i, a_2^j\right) \right\}$$

  - Solved via multiplicative update rule
    - Similarly to GSMM-MAP

Frequency Dependent Separation Algorithm (3)



- Choosing the optimal pair

$$\left(i^*(t), j^*(t)\right) = \underset{i,j}{\operatorname{argmax}} \left\{ \tilde{p}\left(X(f,t) \mid \theta_1^i, \theta_2^j, \hat{a}_1^i(t), \hat{a}_2^j(t)\right) \cdot \Pr\left(\theta_1^i\right) \cdot \Pr\left(\theta_2^j\right) \right\}$$

Frequency Dependent Separation Algorithm (4)

- Wiener filtering

$$\hat{S}_1(f,t) = \frac{\hat{a}_1^{i^*}(t)\cdot\varphi_1^{i^*}(f)}{\hat{a}_1^{i^*}(t)\cdot\varphi_1^{i^*}(f) + \hat{a}_2^{j^*}(t)\cdot\varphi_2^{j^*}(f)}\cdot X(f,t)$$

Separation Cost Function (1)

- GMM MAP criterion

$$\left(i^*, j^*\right) = \underset{i,j}{\operatorname{argmax}} \left\{ p\left(x \mid \theta_1^i, \theta_2^j, \hat{a}_1^i, \hat{a}_2^j\right) \cdot \Pr\left(\theta_1^i\right) \cdot \Pr\left(\theta_2^j\right) \right\}$$

ML term **Priors**

- ML term
  - minimize - $D_{IS}\left(P_x\left(f,t\right), P_{1+2}\left(f,t\right)\right)$
- Priors
  - Prior probability for each CB entry
  - Sources are statistically independent

Separation Cost Function (2)

- Are these priors sufficient?
  - Hint - **[Benaroya & Bimbot, 2003]**

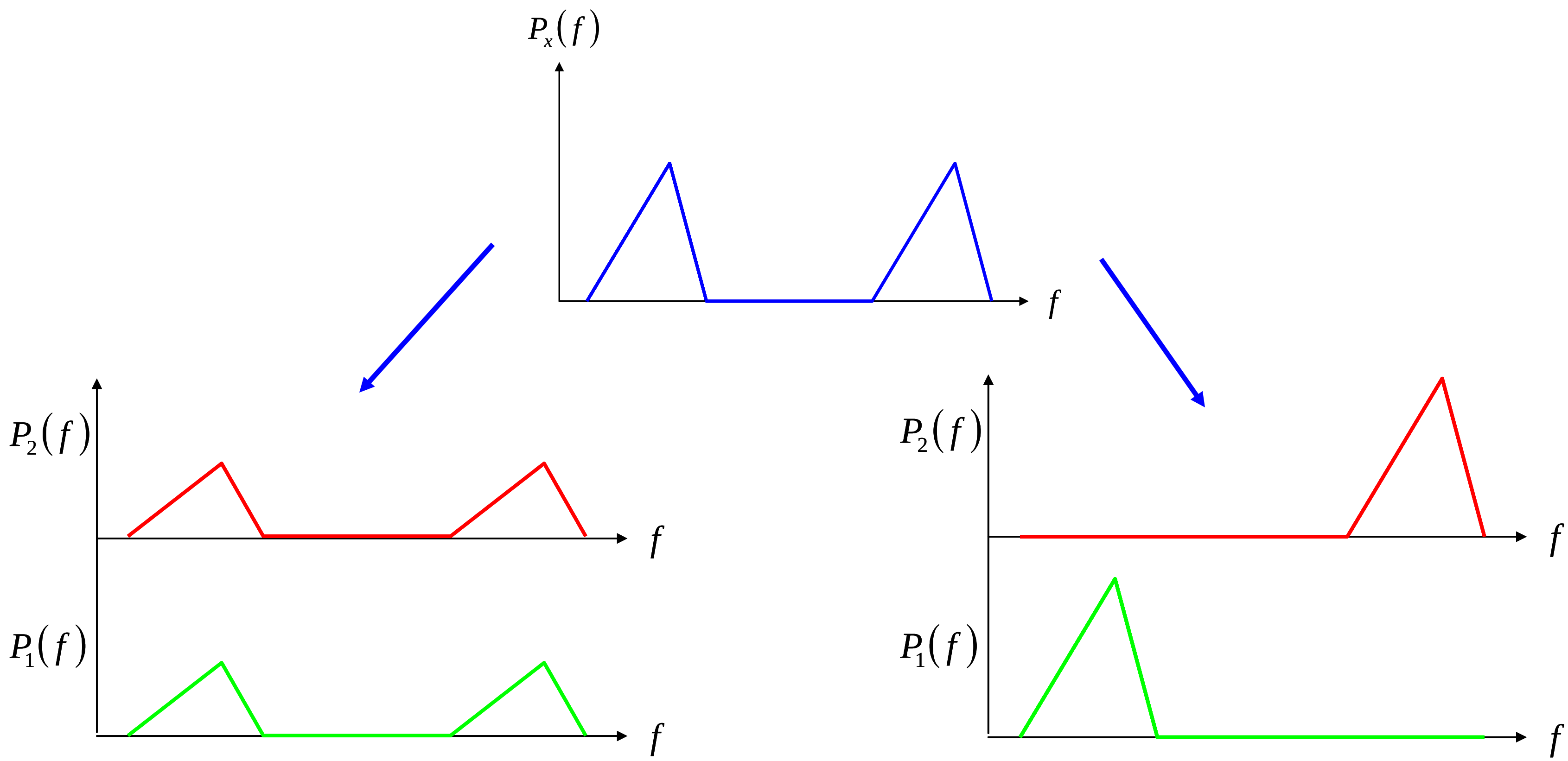
    Using de-correlation as post-processing for improved separation result
- Actual separation

$$D_{IS}\left(P_x\left(f,t\right),P_{1+2}\left(f,t\right)\right)$$

  - Aims to match the observed PSD - $P_x\left(f,t\right)$ with the sources' estimated PSD - $P_{1+2}\left(f,t\right)$

Separation Cost Function (3)

- Which is better <u>separated</u>?



Separation with Distant PSDs Prior (1)

- Introducing an additional prior –

$$\left(i^*, j^*\right) = \underset{i,j}{\operatorname{argmax}} \left\{ p\left(x \mid \theta_1^i, \theta_2^j, \hat{a}_1^i, \hat{a}_2^j\right) \cdot \underbrace{p\left(P_1(f,t), P_2(f,t)\right)}_{\text{New Prior}} \cdot \Pr\left(\theta_1^i\right) \cdot \Pr\left(\theta_2^j\right) \right\}$$

- Intention -
  - The separated signals should be as ‘distant’ as possible
  - Compare the estimated PSDs of the sources
    - Disregard similar PSDs

Separation with Distant PSDs Prior (2)

- $p\left(P_1(f,t),P_2(f,t)\right)$:
  - High probability for distant PSDs
  - Low probability for similar PSDs
- Examples –
  - L_2 prior:

$$p\left(P_1(f,t),P_2(f,t)\right)\propto \exp\left[\frac{\gamma}{2}\left\|P_1(f,t)-P_2(f,t)\right\|_2^2\right]$$

  - Itakura-Saito prior:

$$p\left(P_1(f,t),P_2(f,t)\right)\propto \exp\left[\gamma\cdot\frac{F}{2}\cdot D_{IS}\left(P_1(f,t),P_2(f,t)\right)\right]$$

Separation with Distant PSDs Prior (3)

- Cost function alteration (Gain estimation)
  - L_2 prior:

$$\left(\hat{a}_1^i(t),\hat{a}_2^j(t)\right)=\underset{\left(a_1^i,a_2^j\right)\geq 0}{\operatorname{argmin}}\left\{D_{IS}\left(P_x(f,t),P_{1+2}(f,t)\right)-\frac{\gamma}{F}\left\|P_1(f,t)-P_2(f,t)\right\|_2^2\right\}$$

  - Itakura-Saito prior:

$$\left(\hat{a}_1^i(t),\hat{a}_2^j(t)\right)=\underset{\left(a_1^i,a_2^j\right)\geq 0}{\operatorname{argmin}}\left\{D_{IS}\left(P_x(f,t),P_{1+2}(f,t)\right)-\gamma\cdot D_{IS}\left(P_1(f,t),P_2(f,t)\right)\right\}$$

    - γ - Lagrange multiplier
    - $\gamma=0$ - back to regular GSMM

Distant PSDs Prior Separation Algorithm (1)

Following the cost function alteration

- New separation algorithm evolves
  - With the Itakura-Saito prior
  - Based on GSMM-MAP separation method
  - Can also be applied to AR-based separation methods
- Cost function
  - Match Mixture's PSD to the sources' PSDs
  - Favor distant sources' PSDs

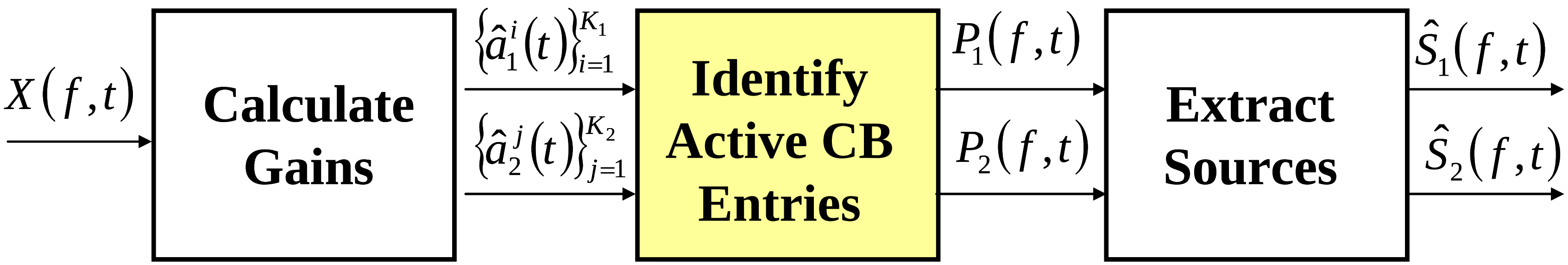
Distant PSDs Prior Separation Algorithm (2)

- ML criterion

$$\left(\hat{a}_1^i(t),\hat{a}_2^j(t)\right)=\underset{\left(a_1^i,a_2^j\right)\geq 0}{\operatorname{argmin}}\left\{D_{IS}\left(P_x(f,t),P_{1+2}(f,t)\right)-\gamma\cdot D_{IS}\left(P_1(f,t),P_2(f,t)\right)\right\}$$

  - Solved via gradient descent algorithm

Distant PSDs Prior Separation Algorithm (3)



- Choosing the optimal pair

$$\left(i^*, j^*\right) = \underset{i,j}{\operatorname{argmax}} \left\{ p\left(x \mid \theta_1^i, \theta_2^j, \hat{a}_1^i, \hat{a}_2^j\right) \cdot p\left(P_1(f,t), P_2(f,t)\right) \cdot \Pr\left(\theta_1^i\right) \cdot \Pr\left(\theta_2^j\right) \right\}$$

Distant PSDs Prior Separation Algorithm (4)

- Wiener filtering

$$\hat{S}_1(f,t)=\frac{\hat{a}_1^{i^*}(t)\cdot\varphi_1^{i^*}(f)}{\hat{a}_1^{i^*}(t)\cdot\varphi_1^{i^*}(f)+\hat{a}_2^{j^*}(t)\cdot\varphi_2^{j^*}(f)}\cdot X(f,t)$$

Experimental Study

- Simulation on real audio signals
- Two separation experiments
  - Speech (TIMIT) & piano (from the web)
  - Speech (TIMIT) & drums (from the web)

Experimental Study

- Distortion measures
- Scenario
- Comparing GMM/AR/NMF
- Comparing GMM to our extensions

Conclusion

- Summary
- Future directions

NMF-based BSS Extensions (1)

- Continuity priors
  - **[Smargadis, 2007]**

    Introducing convolutive NMF for incorporating time dependencies into the NMF framework
  - **[Virtanen, 2007]**

    Introducing continuity constraints on the gain factors into the NMF cost function
- Sparsity priors
  - **[Virtanen, 2007]**

    Introducing Sparsity priors on the gain matrix into the NMF cost function
  - **[Virtanen, 2008]**

    Using the Sparse NMF (SNMF) for single channel source separation

NMF-based BSS Extensions (2)

- Complex NMF
  - **[Kameoka et al., 2009; King & Atlas 2010]**

    Working on the STFT domain, instead of directly on the PSDs
- CASA-driven NMF
  - **[Virtanen, 2007]**

    Weighting frequency bins according to the loudness perception
  - **[Kirbiz & Gunsel 2010]**

    Pre-emphasizing frequency bands that are important for the Human Auditory System (HAS)

GMM-based BSS Extensions (1)

- Several suggested alterations
  - MMSE approach **[Benaroya et al., 2006]**
    - Similar to the MAP estimator (but more complicated)
    - All pairs are participating in the separation stage
      - Using a weighted combination of Wiener filters

$$S_1(f,t)=\sum_{i,j} p\left(\theta_1^i,\theta_2^j \mid X(f,t)\right)\cdot\frac{\hat{a}_1^i(t)\cdot\varphi_1^i(f)}{\hat{a}_1^i(t)\cdot\varphi_1^i(f)+\hat{a}_2^j(t)\cdot\varphi_2^j(f)}\cdot X(f,t)$$

    - Separation quality – ~identical to MAP

GMM-based BSS Extensions (2)

- **[Benaroya et al., 2003]**

  Using Hidden Markov Model (HMM) in order to describe time-correlation between adjacent frames

- **[Ozerov et al., 2005,2007]**

  On-line update of the sources' CBs using EM for voice/music separation (Requires VAD)

- **[Abramson & Cohen, 2008]**

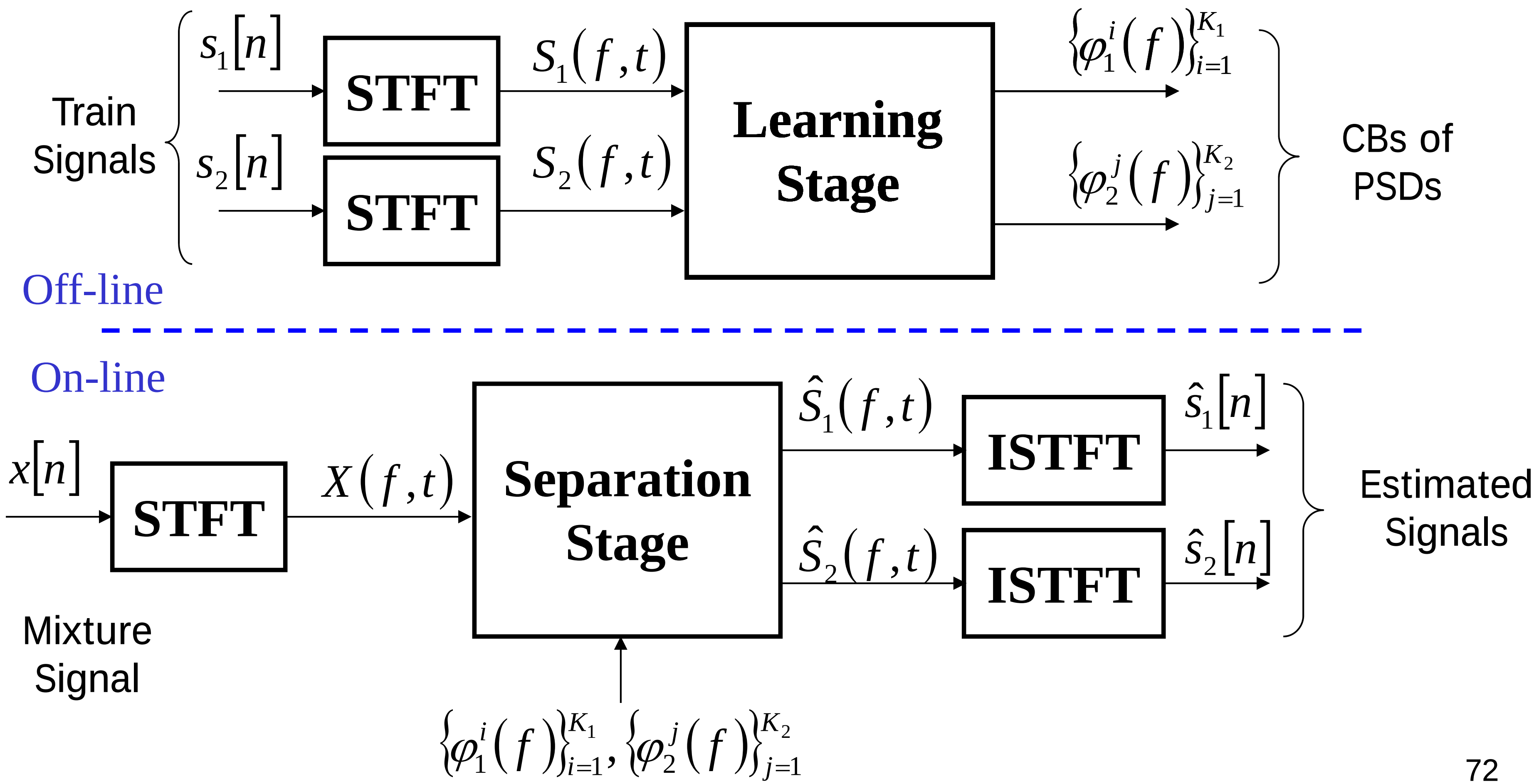
  Introducing a classification and estimation approach on-top the GMM-based separation method

- **[Emiya et al., 2009]**

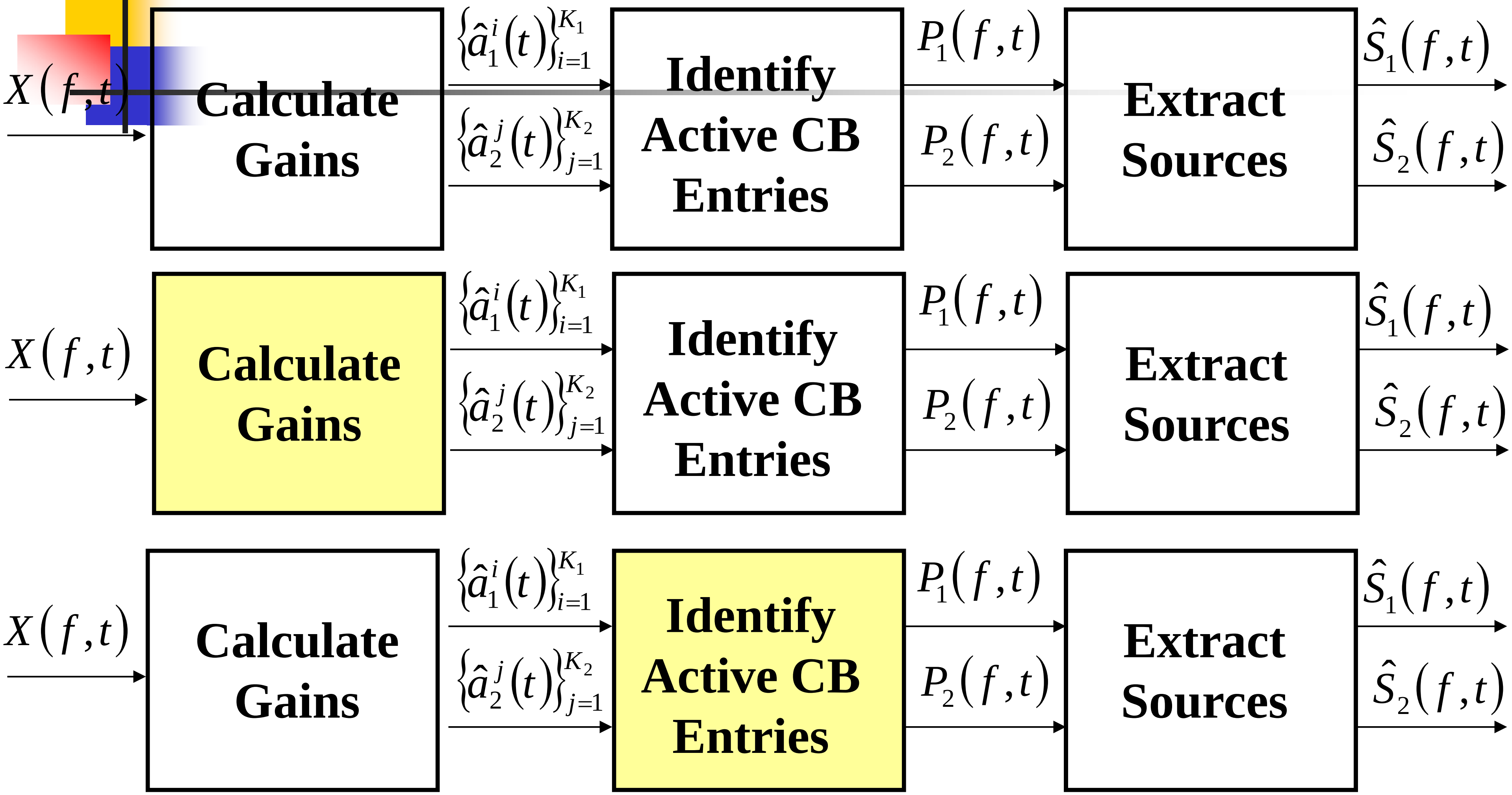
  Learning a CB of the mixture instead of the sources

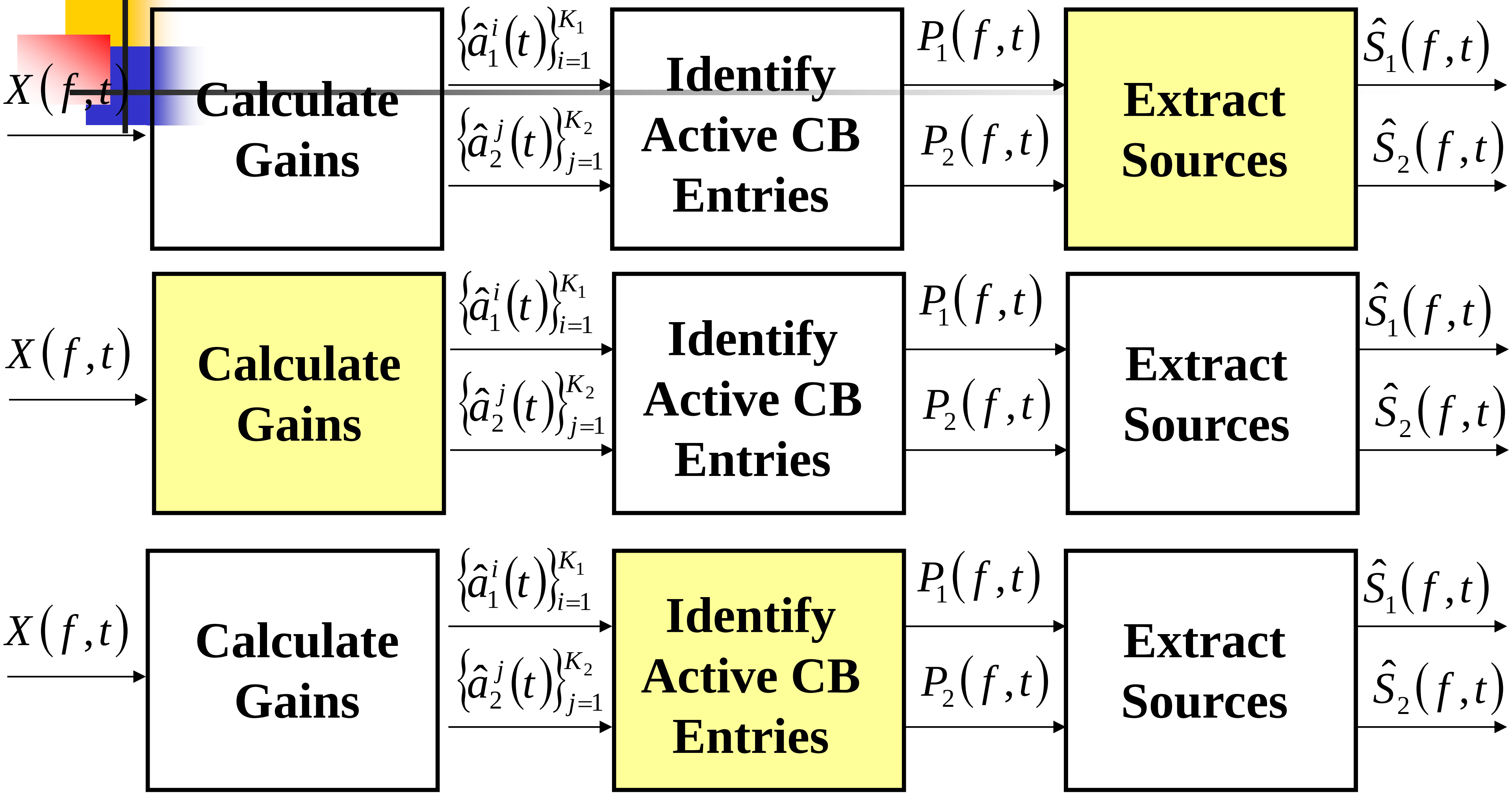
- **[Litvin & Cohen, 2010]**

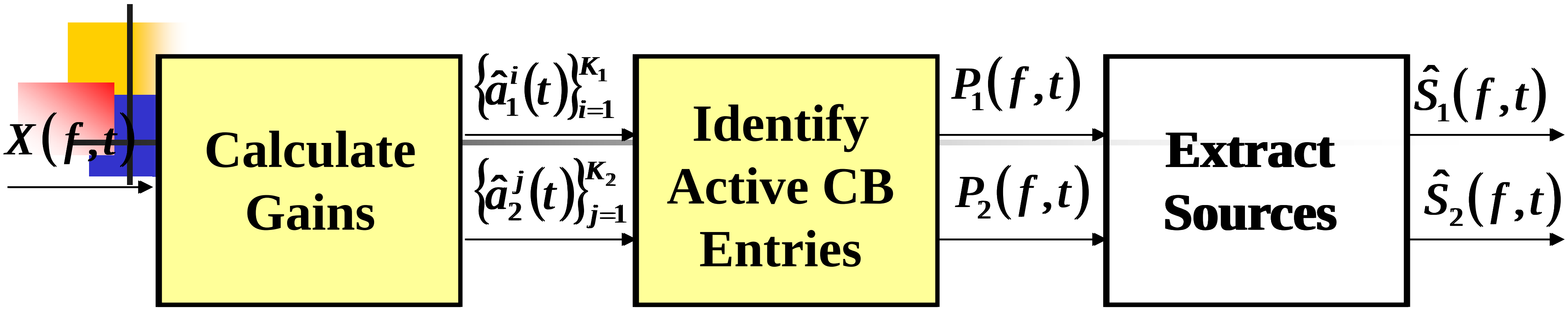
  Working in the Bark-scale wavelet domain instead of STFT

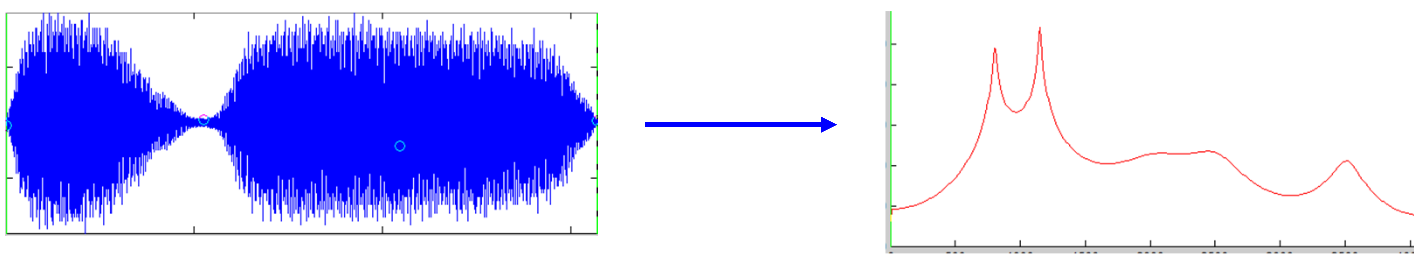


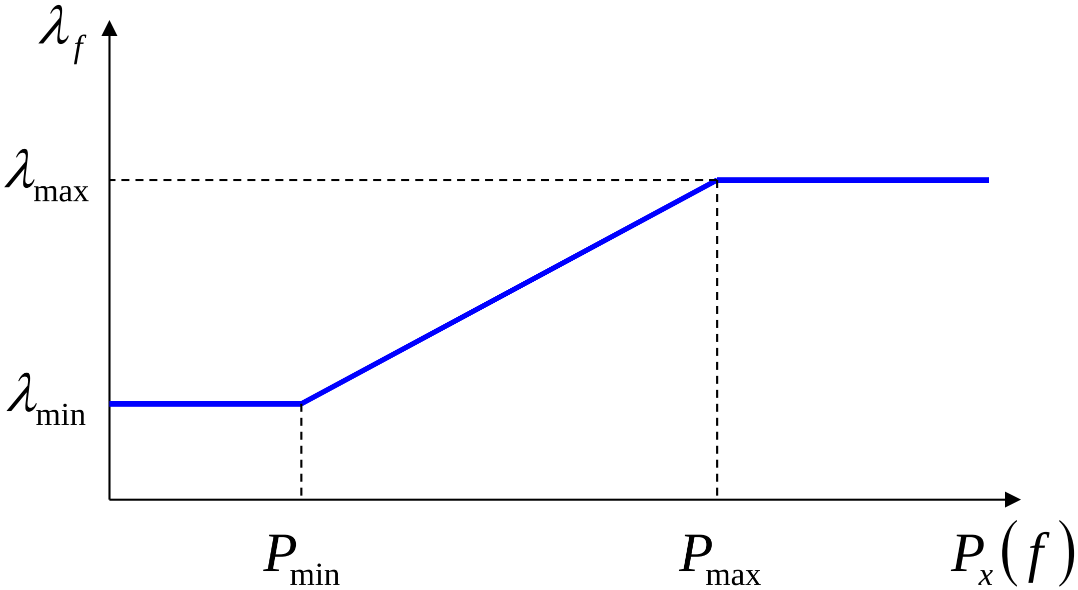
Train Signals
$s_1[n]$
$s_2[n]$
STFT
STFT
$S_1(f,t)$
$S_2(f,t)$
Learning Stage
$\{\varphi_1^i(f)\}_{i=1}^{K_1}$
$\{\varphi_2^j(f)\}_{j=1}^{K_2}$
CBs of PSDs
Off-line
On-line
$x[n]$
STFT
Mixture Signal
$X(f,t)$
Separation Stage
$\hat{S}_1(f,t)$
$\hat{S}_2(f,t)$
ISTFT
ISTFT
$\hat{s}_1[n]$
$\hat{s}_2[n]$
Estimated Signals
$\{\varphi_1^i(f)\}_{i=1}^{K_1}, \{\varphi_2^j(f)\}_{j=1}^{K_2}$

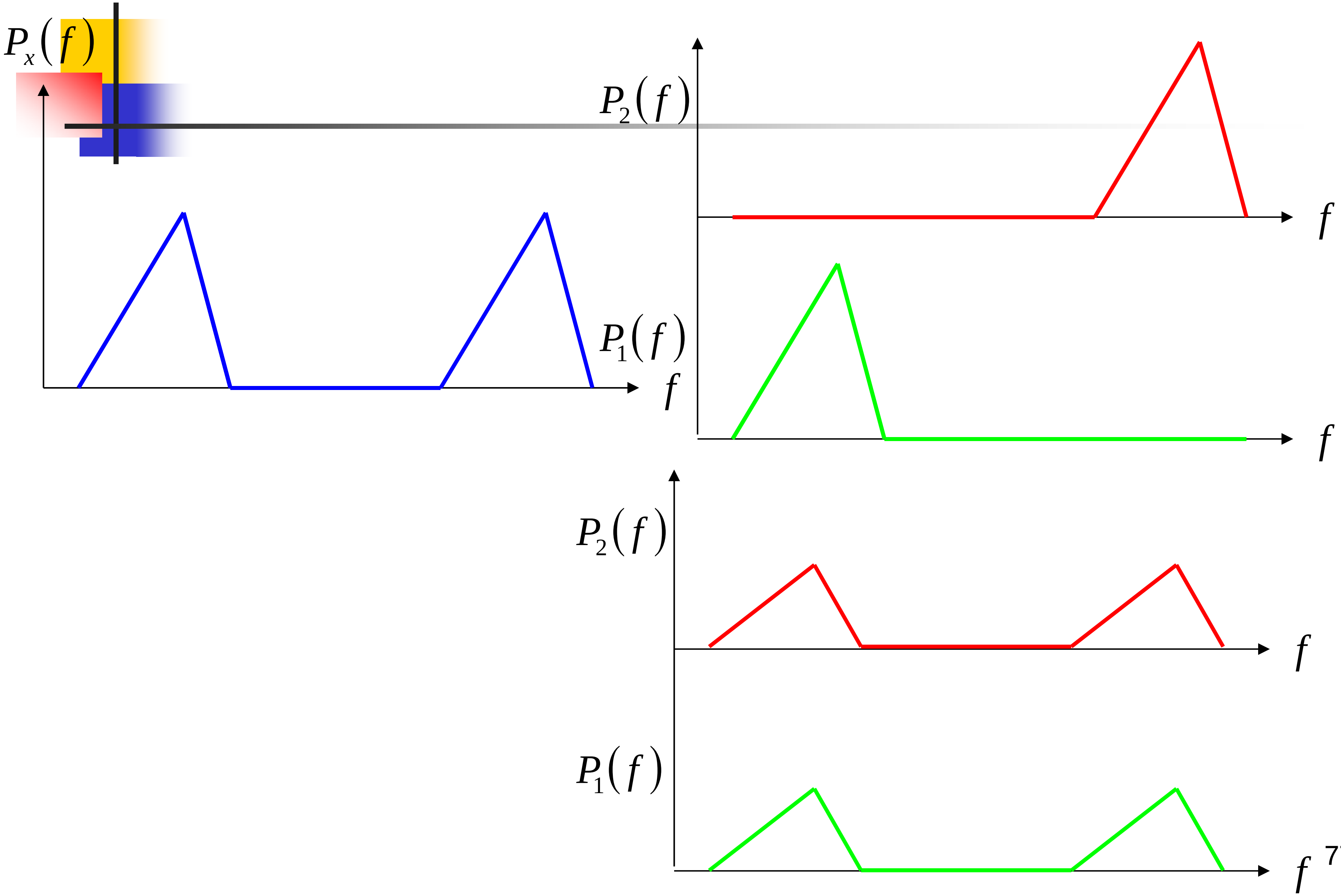

$X(f,t)$
Calculate Gains
$\{\hat{a}_1^i(t)\}_{i=1}^{K_1}$
$\{\hat{a}_2^j(t)\}_{j=1}^{K_2}$
Identify Active CB Entries
$P_1(f,t)$
$P_2(f,t)$
Extract Sources
$\hat{S}_1(f,t)$
$\hat{S}_2(f,t)$


$X(f,t)$
Calculate Gains
$\{\hat{a}_1^i(t)\}_{i=1}^{K_1}$
$\{\hat{a}_2^j(t)\}_{j=1}^{K_2}$
Identify Active CB Entries
$P_1(f,t)$
$P_2(f,t)$
Extract Sources
$\hat{S}_1(f,t)$
$\hat{S}_2(f,t)$


$X(f,t)$
Calculate Gains
$\{\hat{a}_1^i(t)\}_{i=1}^{K_1}$
$\{\hat{a}_2^j(t)\}_{j=1}^{K_2}$
Identify Active CB Entries
$P_1(f,t)$
$P_2(f,t)$
Extract Sources
$\hat{S}_1(f,t)$
$\hat{S}_2(f,t)$




λ_f
$\lambda_{\max}$
$\lambda_{\min}$
$P_{\min}$
$P_{\max}$
$P_x(f)$


$P_x(f)$
f
$P_2(f)$
f
$P_1(f)$
f
$P_2(f)$
f
$P_1(f)$
f