Trabajo Fin de Grado
Grado en Ingeniería de las Tecnologías de Telecomunicación

Separación de audio con modulación

Autor: Antonio Márquez Tristán

Tutor: Iván Durán Díaz

Dep. Teoría de la Señal y Comunicaciones
Escuela Técnica Superior de Ingeniería
Universidad de Sevilla

Sevilla, 2018

Departamento de Teoría de la Señal y Comunicaciones

Trabajo Fin de Grado
Grado en Ingeniería de las Tecnologías de Telecomunicación

Separación de audio con modulación

Autor:
Antonio Márquez Tristán

Tutor:
Iván Durán Díaz
Profesor Titular

Dep. Teoría de la Señal y Comunicaciones
Escuela Técnica Superior de Ingeniería
Universidad de Sevilla

Sevilla, 2018

Trabajo Fin de Grado: Separación de audio con modulación

Autor: Antonio Márquez Tristán
Tutor: Iván Durán Díaz

El tribunal nombrado para juzgar el trabajo arriba indicado, compuesto por los siguientes profesores:

Presidente:

Vocal/es:

Secretario:

acuerdan otorgarle la calificación de:

El Secretario del Tribunal

Fecha:

Agradecimientos

Este trabajo representa el fin de una de las etapas más emocionantes de mi vida. Tras un inicio universitario muy duro, en el cual estuve a punto de tirar la toalla, me causa tremenda alegría verme redactando estas líneas. Han sido muchas las personas que me han apoyado desde el principio y han hecho que haya podido vencer todos los momentos poco gratificantes que esta carrera me ha brindado, incluso cuando los resultados no eran muy halagüeños, ellos han confiado en mis posibilidades, en ocasiones, incluso más que yo mismo. Llegados a este punto solo me queda agradecerles su confianza, ya que para mí es todo un orgullo poner fin a esta etapa y verme totalmente capacitado para enfrentarme a la siguiente.

De todas esas personas tengo que destacar a mis padres, Manolo y Antonia, que son para mí un modelo a seguir y el mayor apoyo que he tenido durante estos años. A mis hermanos, Manolo y Andrés, que también han confiado ciegamente en mí y que han financiado buena parte de mi vida de ocio durante la carrera. Y a mis amigos de toda la vida: Joaquín, Pepe, Jesuli, Sergio y Jero. Sumamos etapas siendo siempre un apoyo los unos para los otros aún cuando por motivos académicos hemos podido vernos menos que en otras épocas.

A todas las buenas personas que he conocido en la escuela, sin ellos toda esta lucha hubiera sido mucho más dura. En especial, a mi grupo de amigos *"Los Peppers"*, sin vosotros mis futuros recuerdos de la universidad hubieran estado faltos de buenos ratos, de cante y palmas después de esas largas jornadas de estudio.

A Mari Ángeles, mi compañera y la que ha tenido que aguantar todos mis malos momentos durante la ejecución de este proyecto, sin tu incondicional apoyo y tu cariño durante estos meses, no hubiera sido lo mismo.

Y por último y no menos importante, agradecer a mi tutor Iván Durán Díaz que me haya dado la posibilidad de hacer este proyecto que tan interesante me ha resultado, y que me haya ayudado a poder realizarlo con éxito. Al señor Fabian-Robert Stöter, por resolverme varias dudas sobre su trabajo y por compartir amablemente material imprescindible para la resolución de este proyecto. También es justo agradecer a todos los profesores que han compartido sus conocimientos conmigo y mis compañeros durante estos años.

Antonio Márquez Tristán
Sevilla, 2018

Resumen

En este trabajo se presenta una solución para el problema de Separación Ciega de Fuentes, especialmente para el caso de fuentes de audio unísonas y moduladas, tanto en tiempo como en frecuencia mediante un vibrato. A partir de una base teórica sobre el problema y sobre el método para solucionarlo, la Factorización No Negativa de Matrices, se expone un modelo de forma teórica y práctica y su posible ejecución en Matlab ® mediante un algoritmo.

El caso en el que se centra este trabajo, el de dos fuentes moduladas y unísonas, representa un gran desafío ya que con NMF no se pueden asumir ciertas suposiciones de importante relevancia para una correcta separación.

En la búsqueda de una solución eficiente, vamos a trabajar con tensores, haciendo uso de la Factorización No Negativa de Tensores, que es una ampliación de NMF a tensores. Dividiremos la STFT de la señal de audio original en parches solapados y calcularemos la 2D-DFT a cada parche, obteniendo así un tensor de 4 dimensiones. A esto es a lo que se ha llamado Transformada de Destino Recurrente, que se incluye en el Modelo de Destino Recurrente.

Sobre este tensor aplicaremos el algoritmo multiplicativo de separación, lo que marca la diferencia frente a NMF, que lo aplica sobre la STFT.

Obtenida la separación, resulta necesario evaluar la calidad de la misma, para lo que se ha usado la herramienta *BSS Eval*.

Después se ha hecho un estudio sobre la dependencia del algoritmo a los parámetros alfa y beta. Se han buscado valores óptimos de separación en función de estos parámetros.

Por último, se presentan diversas simulaciones con las que se ha buscado comprobar las ventajas de este modelo y sus posibles carencias, así como sus posibles líneas futuras y las conclusiones que hemos obtenido tras el trabajo realizado.

Abstract

In this paper we present a solution for the Blind Source Separation problem, especially in the case of unison and modulated audio sources, both in time and frequency because of a vibrato. From a theoretical base on the problem and on the method to solve it, the Non-Negative Matrix Factorization, a model is exposed in a theoretical and practical way, and its possible execution in Matlab through an algorithm.

The case in which we have focus this work, the two unisons and modulated sources, has represented a great challenge because NMF can't assume certain assumptions, specially important for a correct separation.

In the search of an efficient solution, we have worked with tensors, using the Non-Negative Tensor Factorization, which is an ampliaton of NMF to tensors. We will divide the original audio signal STFT in overlapped patches and then, 2D-DFT should be compute to every patch, obtaining a 4 dimensions tensor. This procces has been called Common Fate Transform, included in the Common Fate Model.

A multiplicative algorithm will be applied to the computed tensor, this marks the difference with NMF, which apply this algorithm directly to the whole STFT.

Once the separation is obtained, it is necessary to evaluate the quality of the separation, for which the tool *BSS Eval* has been used.

Then, a study on the dependences of the algorithm on the alpha and beta parameters was made. Optimal separation values have been sought as a function of these parameters.

Finally, some different simulations are presented, with these simulations we have tried to check the advantages of this model, and their possibles lacks, as well as its possible future lines y and the conclusions we have obtained after the work done.

Índice Abreviado

Índice

Notación

$\mathbb{R}$	Cuerpo de los números reales
$\mathbb{C}$	Cuerpo de los números complejos
$\|\mathbf{v}\|$	Norma del vector $\mathbf{v}$
$\langle \mathbf{v}, \mathbf{w} \rangle$	Producto escalar de los vectores $\mathbf{v}$ y $\mathbf{w}$
$\|\mathbf{A}\|$	Determinante de la matriz cuadrada $\mathbf{A}$
$\det(\mathbf{A})$	Determinante de la matriz (cuadrada) $\mathbf{A}$
$\mathbf{A}^\top$	Transpuesto de $\mathbf{A}$
$\mathbf{A}^{-1}$	Inversa de la matriz $\mathbf{A}$
$\mathbf{A}^\dagger$	Matriz pseudoinversa de la matriz $\mathbf{A}$
$\mathbf{A}^{\mathrm{H}}$	Transpuesto y conjugado de $\mathbf{A}$
$\mathbf{A}^*$	Conjugado
c.t.p.	En casi todos los puntos
c.q.d.	Como queríamos demostrar
■	Como queríamos demostrar
□	Fin de la solución
e.o.c.	En cualquier otro caso
e	número e
$\mathrm{e}^{\mathrm{j}x}$	Exponencial compleja
$\mathrm{e}^{\mathrm{j}2\pi x}$	Exponencial compleja con 2π
$\mathrm{e}^{-\mathrm{j}x}$	Exponencial compleja negativa
$\mathrm{e}^{-\mathrm{j}2\pi x}$	Exponencial compleja negativa con 2π
$\mathbb{R}\mathrm{e}$	Parte real
$\mathbb{I}\mathrm{m}$	Parte imaginaria
sen	Función seno
tg	Función tangente
arc tg	Función arco tangente
$\sin^y x$	Función seno de x elevado a y
$\cos^y x$	Función coseno de x elevado a y
Sa	Función sampling
sgn	Función signo
rect	Función rectángulo
Sinc	Función sinc
$\frac{\partial y}{\partial x}$	Derivada parcial de y respecto a x
x°	Notación de grado, x grados.
$\Pr(A)$	Probabilidad del suceso A
$E[X]$	Valor esperado de la variable aleatoria X
σ_X^2	Varianza de la variable aleatoria X
$\sim f_X(x)$	Distribuido siguiendo la función densidad de probabilidad $f_X(x)$
$\mathcal{N}(m_X, \sigma_X^2)$	Distribución gaussiana para la variable aleatoria X, de media m_X y varianza σ_X^2

$\mathbf{I}_n$	Matriz identidad de dimensión n
$\mathrm{diag}(\mathbf{x})$	Matriz diagonal a partir del vector $\mathbf{x}$
$\mathrm{diag}(\mathbf{A})$	Vector diagonal de la matriz $\mathbf{A}$
SNR	Signal-to-noise ratio
MSE	Minimum square error
$:$	Tal que
$\stackrel{\mathrm{def}}{=}$	Igual por definición
$\lVert\mathbf{x}\rVert$	Norma-2 del vector $\mathbf{x}$
$\lvert\mathbf{A}\rvert$	Cardinal, número de elementos del conjunto $\mathbf{A}$
$\mathbf{x}_i, i = 1,2,\ldots,n$	Elementos i, de 1 a n, del vector $\mathbf{x}$
$\mathrm{d}x$	Diferencial de x
$\leqslant$	Menor o igual
$\geqslant$	Mayor o igual
$\backslash$	Backslash
$\Leftrightarrow$	Si y sólo si
$x = a+3 \underset{\substack{\uparrow \\ a=1}}{=} 4$	Igual con explicación
$\frac{a}{b}$	Fracción con estilo pequeño, a/b
Δ	Incremento
$b \cdot 10^a$	Formato científico
$\underset{x}{\rightarrow}$	Tiende, con x
O	Orden
™	Trade Mark
$\mathbb{E}[x]$	Esperanza matemática de x
$\mathbf{C_x}$	Matriz de covarianza de $\mathbf{x}$
$\mathbf{R_x}$	Matriz de correlación de $\mathbf{x}$
σ_x^2	Varianza de x

1 Introducción

1.1 Motivación del Proyecto

La separación de señales de audio ha sido un campo muy prolífico de estudio en los últimos 30 años. Aunque para muchas situaciones existen algoritmos conocidos que proporcionan muy buenos resultados, todavía hay casos límite, con condiciones extremas, en los que existe mucho margen de mejora y se debe seguir investigando. El caso en concreto que nos ocupa, es la separación de dos fuentes mono-canal, unísonas y moduladas tanto en amplitud como en frecuencia, esto no es más que dos instrumentos tocando la misma nota mientras ejecutan un vibrato.

En este trabajo, nos centraremos exclusivamente en el área del procesamiento digital de señales enfocada a la separación de señales de audio, que se engloba dentro del problema de Separación Ciega de Fuentes (Blind Source Separation, BSS) y empleamos la Factorización de Matrices No Negativas (Nonnegative Matrix Factorization, NMF), especialmente su extensión al uso de tensores (NTF), para resolver el problema. Las técnicas de BSS son aplicables a multitud de problemas como: el tratamiento de imágenes médicas, de vídeo, comunicaciones, etcétera.

La separación de fuentes de audio continúa siendo un campo de investigación muy activo desde que se empezara a investigar a principios de la década de 1980. Se han desarrollado diversos métodos de separación, que explotan diferentes características de las señales, entre ellos se puede usar NMF, que factoriza la matriz de un espectrograma en el producto de dos matrices, una llamada de frecuencia y otra de activación, haciendo posible diseñar fácilmente algoritmos eficientes que buscan minimizar la diferencia entre la matriz o tensor original y el producto matricial o tensorial de sus componentes. Para ello, se busca la divergencia y el tipo de actualización óptimos para cada algoritmo. Al mismo tiempo, aporta una reducción de rango, necesaria para descomponer mezclas en sus componentes asociadas a las fuentes. Aplicando los conceptos de NMF a los tensores se pudieron desarrollar modelos más complejos, útiles en muchas aplicaciones, como la separación multi-canal [43]. Algunos de los casos particulares de NMF, como el convolutivo o NMF invariante en el tiempo, también se han aplicado a los algoritmos de NTF. Estos enfoques, aplicados a la descomposición de mezclas de instrumentos musicales, funcionan cuando determinadas suposiciones son ciertas. Una es que los armónicos espectrales solo se solapan parcialmente. Sin embargo, cuando dos fuentes comparten la misma frecuencia fundamental, la mayoría de los armónicos se solapan, reduciendo así el porcentaje de éxito de los algoritmos basados en NMF en el aprendizaje de matrices únicas. Otra suposición es que todas las matrices temporales y espectrales semánticamente corresponden a notas musicales, formando un diccionario de átomos con sentido musical. Esto no se cumple para instrumentos con fluctuaciones variables en el tiempo. Estos efectos se pueden encontrar en instrumentos como los de cuerda o los de viento-metal cuando tocan con vibrato. En el caso en el que dos instrumentos tocan con vibrato la misma nota, las dos suposiciones anteriores pueden no cumplirse, lo que convierte este escenario en un desafío [54]. En vez de aumentar el número de plantillas por fuente, *Hennequin* propone [26] usar matrices de activación dependientes en frecuencia mediante el uso de un modelo basado en fuente/filtro. Como el vibrato no solo causa modulaciones en frecuencia (FM), si no que también causa modulaciones de amplitud (AM), esto recibe el nombre de espectros de modulación, que pueden ser usados para identificar el patrón de modulación. Estos espectros, a veces, son calculados aplicando la transformada de Fourier a un espectro de magnitud. El *spectrograma de*

modulación ya ha captado mucha atención en el campo del reconocimiento [23][30] y clasificación de voz [31] [39]. *Barker y Virtanen* [7] fueron los primeros en proponer una modulación representada con tensores para una separación de fuentes monocanal. Esto permite aplicar la factorización al tensor usando la conocida descomposición CANDECOMP/PARAFAC (CP) [25].

1.2 Objetivo del trabajo

El objetivo de este trabajo es el estudio de la separación ciega de fuentes de audio para el caso concreto de una mezcla mono-canal de dos instrumentos que realizan un vibrato mientras tocan ambos la misma nota, para cuya resolución, haremos uso del método conocido como Factorización No Negativa de Tensores (NTF). Partiendo de los resultados presentados por *Stöter et al.* en [53], se va a hacer un estudio de los valores de los parámetros alfa y beta para los que se obtienen mejores resultados en la separación.

El método de descomposición tensorial empleado en [53] explota las similitudes en frecuencia. También nos permite hacer uso de las dependencias entre las modulaciones de los intervalos vecinos. Esto tiene ciertas coincidencias con el modelo HR-NMF, del cual se habla en la Sección 2.2.4 y que tiene en cuenta las dependencias en el plano tiempo-frecuencia. El método propuesto en [53], relaja algunas suposiciones tomadas en HR-NMF con la intención de simplificar el proceso de estimación.

Por último, se ha hecho un estudio sobre cómo afectan distintos valores de los parámetros alfa y beta, en función del tipo de fuente, a la calidad de los resultados obtenidos por el método propuesto en [53].

En resumen, el objetivo de este proyecto es estudiar una solución al problema de BSS para un caso muy concreto donde algunas suposiciones esenciales de NMF no se cumplen, estimando los parámetros NMF basándonos en la divergencia AB y utilizando un algoritmo MU, que sea capaz de combinar otros algoritmos multiplicativos existentes, de manera que pueda ser aplicado a distintos casos, ajustando el valor de solo dos parámetros.

1.3 Estructura del proyecto

Este documento está dividido en cinco capítulos, cada uno de los cuales consta a su vez de diferentes secciones y subsecciones.

Capítulo 1: corresponde a la introducción del trabajo. Incluye las motivaciones que nos han llevado a hacerlo, el objetivo para el que se ha realizado y una explicación de la estructura de la memoria.

Capítulo 2: en este capítulo se expone el problema a resolver, se definen de manera amplia conceptos generales como la Separación Ciega de Fuentes y su aplicación a las señales de audio, por ser el motivo de nuestro estudio. Finalmente, se ahonda en el método de la Factorización No Negativa de Matrices, en el que está basado el algoritmo que se ha empleado para resolver el problema de separación.

Capítulo 3: se presenta el método utilizado para resolver el problema, propuesto en [53]. En primer lugar, se presenta el modelo de descomposición del espectrograma expuesto en [53], Modelo de Destino Recurrente, después el proceso matemático que nos servirá para calcular la separación, la Transformada de Destino Recurrente, también se detalla el algoritmo MU y cómo estimar sus parámetros. Por último, hay una sección dedicada a explicar el funcionamiento de las divergencias AB y su sentido en este trabajo.

Capítulo 4: dedicado a las simulaciones realizadas en Matlab® y a los resultados obtenidos por éstas. En este capítulo se comenta cómo se ha implementado el algoritmo.

Capítulo 5: finalmente, se presentan las conclusiones y se proponen líneas futuras de trabajo.

2 Técnicas de Separación. Separación de Señales de Audio.

En el campo de la separación de señales, existen numerosas técnicas para separar las señales que se encuentran mezcladas en un conjunto de observaciones. Estas técnicas se agrupan bajo el nombre de Separación Ciega de Fuentes (Blind Source Separation, BSS) y una frecuentemente usada, cuando se cumplen ciertas hipótesis, es la Factorización de Matrices No Negativas (Non-Negative Matrix Factorization, NMF), que es una técnica general de descomposición de observaciones de valores no negativos.

2.1 Separación Ciega de Fuentes (Blind Source Separation, BSS)

En el campo de la física y la ingeniería se llama Separación Ciega de Fuentes (BSS) a la recuperación de señales no observadas o fuentes que se encuentran mezcladas en un conjunto de señales conocidas. El hecho de que las señales de origen no se conozcan y no haya información disponible sobre la mezcla, es el motivo por el que se usa el adjetivo *ciega* en este tipo de separación [9]. Las técnicas de BSS han sido desarrolladas durante las últimas dos décadas; muchos algoritmos han sido desarrollados y aplicados en una amplia gama de aplicaciones que incluyen ingeniería biomédica, imágenes médicas, reconocimiento de voz, imágenes astronómicas y sistemas de comunicación [41].

El problema de la separación de fuentes fue formulado alrededor de 1982 por *Bernard Ans, Jeanny Hérault y Christian Jutten*, en el marco del modelado neuronal, para la decodificación del movimiento de vertebrados. El problema también se ha planteado de forma independiente en el marco de las comunicaciones [6].

Las primeras contribuciones a conferencias de procesamiento de señales y de redes neuronales, aparecieron alrededor de 1985. Inmediatamente, estos documentos llamaron la atención de los investigadores enfocados en el procesamiento de señales, principalmente en Francia y más tarde en Europa. En la comunidad de redes neuronales, el interés surgió mucho más tarde, en 1995, pero de forma muy masiva. Inicialmente, se investigó la separación de fuentes para mezclas lineales instantáneas (sin memoria). A principios de la década de 1990, buena parte de los estudios en el campo ya estaban centrados en las mezclas convolutivas. Finalmente, las mezclas no lineales, excepto unos pocos estudios aislados, se abordaron a finales de la década de 1990 [14].

2.1.1 BSS de mezclas lineales e instantáneas

El modelo de mezclas lineales e instantáneas es el más simple y el problema más fácil de resolver cuando se tiene un número suficiente de sensores. El problema tratado en este trabajo es un problema de mezclas lineales e instantáneas, pero no contamos con un número suficiente de canales para realizar la separación, lo que complica notablemente el proceso.

Dadas J señales desconocidas $s_1(t), s_2(t), \ldots, s_J(t)$, a las que llamaremos fuentes en un modelo de mezclas lineales e instantáneas, cada una de las observaciones $x_1(t), x_2(t), \ldots, x_M(t)$, pueden escribirse como

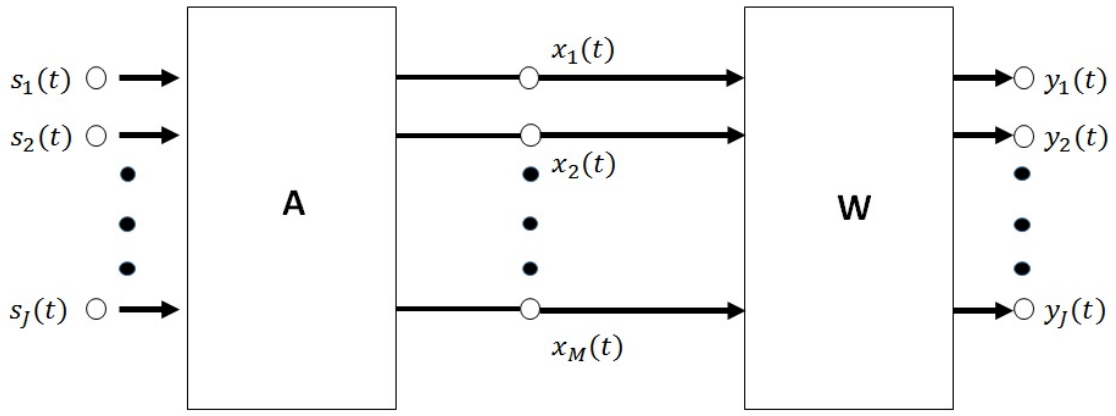

Figura 2.1 Modelo BSS lineal instantáneo [60].

combinación lineal de las fuentes. El modelo se muestra en la Figura 2.1.

$$x_i(t) = \sum_{j=1}^{J} a_{ij} s_j(t) \quad i = 1, \ldots, M \tag{2.1}$$

Los escalares a_{ij} son los coeficientes de mezcla. Agrupando las fuentes en el vector de fuentes $\mathbf{s}(t) = [s_1(t), s_2(t), \ldots, s_J(t)]^T$ y las observaciones en el vector de observaciones $\mathbf{x}(t) = [x_1(t), x_2(t), \ldots, x_M(t)^T$, podemos escribir la ecuación (2.1) de forma matricial como

$$\mathbf{x}(t) = \mathbf{A}\mathbf{s}(t) \tag{2.2}$$

donde $\mathbf{A}$ es la matriz de mezcla de dimensión $M \times J$, cuyos elementos son a_{ij}. Tanto $\mathbf{s}(t)$ como $\mathbf{x}(t)$ pueden contener valores complejos. Para que la mezcla sea invertible, y recuperar el vector de fuentes $\mathbf{s}(t)$, es necesario que el número de filas de $\mathbf{A}$ sea mayor o igual que el de columnas, es decir, que el número de observaciones sea mayor igual que el de fuentes ($M \geq J$). Este modelo es generativo, es decir, describe cómo los datos observados son generados mediante un proceso de mezcla de las componentes s_j.

La idea básica de la separación ciega de fuentes es estimar las señales originales a través de una matriz de separación $\mathbf{W}$, de dimensión $J \times M$, siendo la matriz de mezcla $\mathbf{A}$ y el vector de fuentes $\mathbf{s}(t)$ desconocidos [60].

$$\mathbf{y}(t) = \mathbf{W}\mathbf{x}(t) \tag{2.3}$$

Siendo $\mathbf{y}(t) = [y_1, \ldots, y_J]$ el vector de señales de salida que tratan de estimar las fuentes $\mathbf{s}(t)$. En la Figura 2.1, se representa el proceso de mezcla y separación. Si los sistemas $\mathbf{A}$ y $\mathbf{W}$ pueden representarse por matrices constantes en el tiempo, estamos ante un problema de BSS con mezcla lineal e instantánea.

Puesto que las fuentes y la mezcla son desconocidas, para resolver el problema es necesario utilizar cierta información a priori, en forma de hipótesis. En función de las hipótesis empleadas, se obtienen diferentes criterios de BSS, algunos de los cuales se detallan a continuación. Por otra parte, los algoritmos empleados para optimizar estos criterios pueden ser de procesamiento por bloques o bien algoritmos adaptativos.

- *Análisis de Componentes Principales (Principal Component Analysis, PCA)*: transformación del vector de datos $\mathbf{x}(t)$ en un vector de señales incorreladas, es decir, asumir como hipótesis que las fuentes son incorreladas. Estas señales incorreladas son llamadas componentes principales y se obtienen mediante descomposición en autovectores y autovalores o descomposición en valores singulares. Un uso común del PCA es la reducción de la dimensión de la matriz de datos y así, solo un conjunto de componentes principales se mantienen para preservar la máxima varianza de los datos. En PCA, la unicidad se consigue imponiendo ortogonalidad en la matriz de transformación [62].

- *Análisis de Componentes Independientes (Independent Component Analysis, ICA)*: es una generalización del Análisis de Componentes Principales. Siguiendo las definiciones de los pioneros, *Jutten y Hérault* en 1991 y *Common* en 1994, podemos suponer que, tanto las variables de mezcla como las componentes independientes tienen una media cero: si esto no es cierto, las variables observables x_i siempre se pueden centrar restando la media de la muestra, lo que hace que el modelo tenga una media cero.

  El punto de partida del modelo ICA es la suposición de que las componentes s_i son *estadísticamente independientes*. También se asume que las componentes independientes tienen una distribución no Gaussiana, ya que si hay más de una Gaussiana no hay forma de separar usando independencia, debido a que la mezcla de dos variables Gaussianas independientes puede dar lugar a variables Gaussianas independientes. En el modelo básico no se suponen conocidas estas distribuciones (si se conocen, se simplifica el problema considerablemente). Para simplificar, se puede suponer cuadrada la matriz de mezcla, que es desconocida. En audio, es conocido que las fuentes son independientes y no Gaussianas, así que su aplicación en este campo está muy extendida [28].

  Dentro del ámbito más general del Análisis de Variables Latentes, se dice que las *componentes independientes* (las fuentes) son variables latentes, en el sentido en que no pueden ser directamente observadas.

- *Análisis de Componentes Escasas (Sparse Component Analysis, SCA)*: se asume que las fuentes son escasas, es decir, que las fuentes sean cero con frecuencia. El principio básico del SCA consiste en cuatro pasos:

  1. Aplicar una transformada lineal de dispersión a la mezcla. Una de las transformadas que se suele usar para la dispersión es la STFT. La transformada se usa para dispersar la representación de las fuentes, así la representación de cada fuente tiene sólo algunos coeficientes significativos.

  2. Estimar la matriz de mezcla del gráfico de dispersión. Además del uso del gradiente natural del modelo ICA, un enfoque común en la actualidad es confiar en las técnicas de agrupamiento (clustering), con variantes de K-medias ponderadas. Para que estas técnicas funcionen de forma eficiente, la hipótesis clave es asumir que, como máximo, una fuente contribuye significativamente a cada punto del gráfico de dispersión. En el caso de las fuentes de audio, normalmente se asume que, en el dominio tiempo-frecuencia, la actividad de cada fuente muestra cierta persistencia local dentro de las pequeñas regiones de la distribución tiempo-frecuencia donde son "visibles".

  3. Consiste en estimar la representación de las fuentes basándose en la suposición de dispersión. En un escenario libre se ruido, *Bofill y Zibulevsky* [47] propusieron una estimación que se puede interpretar como de máxima probabilidad, asumiendo que los coeficientes de las fuentes tengan una distribución laplaciana.

  4. Reconstrucción de las fuentes invirtiendo la transformada de dispersión.

  SCA es muy útil a la hora de aislar ruidos y distorsiones, ya que normalmente estos suelen tener un nivel bajo en proporción a la señal de interés.

2.1.2 BSS para mezclas convolutivas

Cuando las fuentes contribuyen a la mezcla con numerosas versiones retardadas se consideran mezclas convolutivas [14]. Esto puede ocurrir en diversas aplicaciones como el audio. Las mezclas de audio en entornos reales, debido a la reverberación, se consideran siempre mezclas convolutivas (además de variantes en el tiempo).

La diferencia entre el modelo de mezcla lineal convolutivo y el instantáneo es que, versiones retrasadas de las fuentes contribuyen a la salida del modelo en momentos dados.

Modelo

En el modelo convolutivo, la matriz de mezcla se sustituye por un sistema MIMO (múltiples entradas y múltiples salidas) lineal e invariante en el tiempo (LTI) con respuesta impulsiva$(\mathbf{A}(n))_{n\in\mathbb{Z}}$. Las señales

de observación son, por tanto, determinadas por las fuentes conforme al siguiente modelo de convolución multicanal:

$$\forall n \in \mathbb{Z} \qquad \mathbf{x}(n) = \sum_{k\in\mathbb{Z}} \mathbf{A}(k)\mathbf{s}(n-k). \tag{2.4}$$

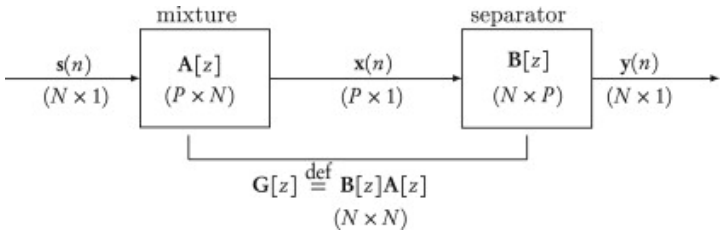

Figura 2.2 Modelo de mezcla convolutiva [14].

Una estructura como la mostrada en la Figura 2.2 puede ser invertida con un sistema MIMO-LTI. Recuperar las fuentes es equivalente a encontrar un sistema MIMO-LTI inverso, llamado separador. Si su respuesta impulsiva se denota por $(\mathbf{B}(n))_{n\in\mathbb{Z}}$, las salidas separadas son dadas por:

$$\forall n \in \mathbb{Z} \qquad \mathbf{y}(n) = \sum_{k\in\mathbb{Z}} \mathbf{B}(k)\mathbf{x}(n-k). \tag{2.5}$$

Debido al contexto convolutivo, debemos usar la transformada Z de los sistemas LTI. Para los sistemas de mezcla y separación, con respuestas impulsivas $(\mathbf{A}(n))_{n\in\mathbb{Z}}$ y $(\mathbf{B}(n))_{n\in\mathbb{Z}}$ respectivamente se define:

$$\mathbf{A}[z] \stackrel{\triangle}{=} \sum_{k\in\mathbb{Z}} \mathbf{A}(k)z^{-k} \qquad y \qquad \mathbf{B}[z] \stackrel{\triangle}{=} \sum_{k\in\mathbb{Z}} \mathbf{B}(k)z^{-k} \tag{2.6}$$

Es conveniente introducir el sistema que combina mezcla y separación. Se obtiene de las ecuaciones 2.5 y 2.6, se aprecia que la salida global en el separador recibe:

$$\forall n \in \mathbb{Z} \qquad \mathbf{y}(n) = \sum_{k\in\mathbb{Z}} \mathbf{G}(k)\mathbf{s}(n-k) \tag{2.7}$$

donde la respuesta impulsiva y la transformada Z del sistema global $(\mathbf{G}(n))_{n\in\mathbb{Z}}$ son dadas por las ecuaciones:

$$\forall n \in \mathbb{Z} \quad \mathbf{G}(n) = \sum_{k\in\mathbb{Z}} \mathbf{G}(n-k)\mathbf{A}(k) \qquad y \qquad \mathbf{G}[z] = \mathbf{B}[z]\mathbf{A}[z]. \tag{2.8}$$

Evolución del modelo y de las técnicas de separación

A continuación, vamos a hablar de algunos modelos específicos dentro de la BSS para mezclas convolutivas en el campo del audio [55]. Antes de introducir dichos modelos, es conveniente aclarar que, el modelo general tiene limitaciones intrínsecas, especialmente para el audio. Primero, el modelado del sistema como respuestas impulsivas entre la localización de cada fuente y la localización de cada micrófono implícitamente asume que, cada fuente emite sonido desde un único punto en el espacio, previniendo así el modelado de fuentes espacialmente difusas. Segundo, a no ser que se conozca información adicional, las fuentes se pueden recuperar, a lo sumo, hasta un filtrado indeterminado. Tercero, el sistema lineal $\mathbf{A}(t)$ puede ser invertido solo en determinados escenarios, en los que el número de fuentes es menor que el de micrófonos ($J \leq I$).

Debido a estas limitaciones, muchos investigadores propusieron enfocar este problema en el dominio del tiempo-frecuencia por medio de la Transformada Localizada de Fourier (STFT) compleja.

En 1998, *Cardoso* [8] propuso reformular el proceso de mezcla como

$$\mathbf{n}(t) = \sum_{j=1}^{J} \mathbf{c}_j(n) \tag{2.9}$$

de forma que el problema de separación de fuente se convirtiera en un problema basado en extraer la contribución $\mathbf{c}_j(t) = [c_{j1}(t),\ldots,c_{jI}(t)]^T$ de cada fuente a la mezcla. Con el tiempo, $\mathbf{c}_j(t)$ fue llamado *imagen*

espacial de la fuente j-ésima [58]. Con esta reformulación se evitó la indeterminación provocada por el filtrado, uniendo $\mathbf{a}_j(t)$ y $s_j(t)$ en una sola cantidad

$$\mathbf{c}_j(t) = \left(\mathbf{a}_j * s_j\right)(n) \tag{2.10}$$

y el modelo general (2.9) se volvió aplicable a fuentes espacialmente difusas, que no puede expresarse como (2.10).

Al mismo tiempo, numerosos investigadores, propusieron pasar el problema al dominio del tiempo-frecuencia, mediante medias de la STFT compleja. Se reformuló el proceso de mezcla en cada cuadro temporal n y en cada intervalo de frecuencia f, de forma que se expresó como:

$$\mathbf{x}(n,f) = \sum_{j=1}^{J} \mathbf{c}_j(n,f), \tag{2.11}$$

En el dominio tiempo-frecuencia, el vector de fuentes se define como $\mathbf{s}(n,f) = [s_1((n,f),\ldots,s_J(n,f)]$ y el vector de observaciones como $\mathbf{x}(n,f) = [x_1(n,f),\ldots,x_m(n,f)]$. El modelo de mezcla convolutivo se aproxima bajo la suposición de banda estrecha, por la multiplicación de valores complejos en cada intervalo de frecuencias

$$\mathbf{c}_j(n,f) = \mathbf{a}_j(f)s_j(n,f), \tag{2.12}$$

donde la transformada de Fourier $\mathbf{a}_j(f)$ de $\mathbf{a}_j(t)$ es el llamado vector de mezcla de la fuente j-ésima o en la forma matricial $\mathbf{x}(n,f) = \mathbf{A}(f)\mathbf{s}(n,f)$, donde $\mathbf{A}(f) = [\mathbf{a}_1(f),\ldots,\mathbf{a}_J(f)]$ es la llamada matriz de mezcla.

La separación de fuentes se reformuló de varias formas, entre ellas, como un problema similar al de agrupación (clustering), por lo que el sonido en un intervalo de tiempo-frecuencia dado debe asignarse a la única o pocas fuentes activas en ese intervalo, y así la separación se hizo viable en escenarios indeterminados, con más fuentes que micrófonos ($J \leq I$) [61]. Otra de estas reformulaciones fue resolver, para cada frecuencia, el problema de la separación, y posteriormente, resolver el de las permutaciones.

Mientras que las primeras técnicas de separación de fuentes se basaban en la diversidad espacial, es decir, en la suposición de que las fuentes tienen diferentes direcciones de llegada, el cambio al dominio del tiempo-frecuencia habilitó la explotación de la diversidad espectral, es decir, la suposición de que sus STFTs seguían distintas distribuciones. Esto posibilitó trabajar con mezclas mono-canal y mezclas de fuentes con la misma dirección de llegada.

En los últimos años se han propuesto importantes mejoras en las técnicas de separación de fuentes de audio cada vez más adecuadas a las propiedades de las fuentes sonoras y a las especificaciones de las mezclas acústicas: numerosos modelos y sofisticados algoritmos se han desarrollado para incorporar información adicional sobre las fuentes o el entorno de la mezcla para guiar el proceso de separación. Estos modelos rompen un poco con las restricciones propias de BSS, por lo que se engloban bajo el término *modelos de separación guiada de fuentes*.

Dentro de estos algoritmos, aquellos que emplean información sobre el comportamiento general de las fuentes de audio y/o del proceso acústico de mezcla, por ejemplo, "las fuentes están escasamente distribuidas" o "la mezcla fue realizada en exterior", se consideran algoritmos *suavemente guiados*. Mientras que los algoritmos que aprovechan información específica sobre la mezcla para la separación, como las posiciones de las fuentes o el género musical, se consideran algoritmos *fuertemente guiados* [55].

Antes de introducir algunos tipos de guía en los algoritmos, es necesario aclarar algunos conceptos comunes de los algoritmos ciegos y guiados. La separación se basa en dos paradigmas de modelado alternativos: la no gaussianidad o no estacionariedad, donde la no estacionariedad se puede manifestar en el tiempo, en frecuencia o en ambos [10]. Estos paradigmas son perfectamente intercambiables: eligiendo uno de ellos no se restringe el tipo de información que se puede incluir como guía o los escenarios prácticos que pueden ser considerados.

- **Modelado No Gaussiano Escaso.**

  Asumiendo que los coeficientes de la STFT de las fuentes siguen una distribución estacionaria no gaussiana $p(.)$, que no es más que su función densidad de probabilidad, la separación se puede lograr en el sentido de máxima verosimilitud como [14]:

$$\min_{A,s} \sum_{j,n,f} -\log p(\mathbf{s}_j(n,f)) \quad \text{sujeto a } \mathbf{x}(n,f) = \mathbf{A}(f)s(s,n). \tag{2.13}$$

  Cuando no se tiene información específica de $\mathbf{A}$ o $\mathbf{s}$, la minimización se consigue restringiendo el escalado, para evitar la divergencia de $\mathbf{A}$ y $\mathbf{s}$ a valores infinitamente grandes o pequeños.

$$\min_{A,s} \frac{1}{2} \sum_{n,f} \| \mathbf{x}(n,f) - \mathbf{A}(f)\mathbf{s}(n,f) \|_2^2 + \lambda \sum_{n,f} \mathscr{P}(\mathbf{s}(n,f)), \tag{2.14}$$

  donde $\mathscr{P}(.)$ es un término de penalización. La elección del parámetro λ no es trivial. Cuando la restricción $\mathbf{x}(n,f) = \mathbf{A}(f)\mathbf{s}(n,f)$ se cumple, el mínimo de $\sum_{n,f} \mathscr{P}(s(n,f))$ sujeto a esta restricción se obtiene para λ próxima a 0.

  Para una longitud de ventana de la STFT típica, del orden de 50-100 ms, los coeficientes de la STFT de señales de audio siguen una distribución escasa, con un pico marcado en cero y colas largas comparadas con la gaussiana. La distribución gaussiana generalizada, $P(\mathbf{s}(n,f)) \propto \exp(-\lambda \mid \mathbf{s}_j(n,f) \mid^p)$ y la norma asociada a la inducción de escasez, $\mathscr{P}(\mathbf{s}(n,f)) = \| \mathbf{s}(n,f) \|_p^p = \sum_{j=1}^J \mid \mathbf{s}_j(n,f) \mid^p$, con $0 < p < 2$, son elecciones populares para modelar este comportamiento.

- **Modelado Gaussiano No Estacionario.**

  Un paradigma alternativo se basa en asumir que los vectores de la STFT de las imágenes espaciales de las fuentes tienen una distribución gaussiana no estacionaria de media cero

$$P\left(\mathbf{c}_j(n,f) \mid \Sigma_{\mathbf{c}_j(n,f)}\right) = \frac{1}{\det\left(\pi \sum_{\mathbf{c}_j}(n,f)\right)} e^{-\mathbf{c}_j(n,f)^H \sum^{-1}_{\mathbf{c}_j(n,f)} \mathbf{c}_j(n,f)} \tag{2.15}$$

  donde ^{H} denota el conjugado traspuesto. La covarianza $\sum_{\mathbf{c}_j}(n,f)$ depende tanto del tiempo como de la frecuencia. Se puede factorizar como el producto de una potencia escalar en el espectro temporal $\upsilon_f(n,f)$ y una matriz de covarianza espacial $\mathbf{R}_f(f)$ [17]

$$\Sigma_{\mathbf{c}_j}(n,f) = \upsilon_f(n,f)\mathbf{R}_f(f). \tag{2.16}$$

  La separación se consigue estimando los parámetros del modelo en el sentido de máxima verosimilitud

$$\min_{\mathbf{R},\upsilon} \sum_{j,n,f} -\log P(\mathbf{c}_j(n,f) \mid \mathbf{R},\upsilon) \quad \text{sujeto a} \quad \mathbf{x}(n,f) = \sum_{j=1}^{J} \mathbf{c}_j(n,f) \tag{2.17}$$

  usando un algoritmo esperanza-maximización (EM). Una vez estimados $\mathbf{R}$ y υ, $\mathbf{c}_j(n,f)$ puede derivarse en el sentido del mínimo error cuadrático medio con un filtrado de Wiener multicanal.

$$\hat{\mathbf{c}}_j(n,f) = \sum_{\mathbf{c}_j}(n,f) \left(\sum_{j=1}^{J} \sum_{\mathbf{c}_j}(n,f) \right)^{-1} \mathbf{x}(n,f). \tag{2.18}$$

Una vez introducidos los dos paradigmas, vamos a exponer algunas de las formas de introducción de información en los modelos, conocidos como guías. Las ecuaciones (2.13), (2.14) y (2.17) forman la base de todos los algoritmos guiados presentados a continuación. Sin información sobre $\mathbf{A}$, $\mathbf{s}$, $\mathbf{R}$ o υ, la imagen espacial de la fuente $\mathbf{c}_j(n,f)$ puede recuperarse, en el mejor de los casos, hasta una permutación arbitraria en cada intervalo de frecuencia. Este supuesto problema de permutación fue históricamente el primer motivo para investigar la incorporación de más información a los modelos.

La información puede introducirse ya sea en forma de restricciones deterministas sobre $\mathbf{A}$, $\mathbf{s}$, $\mathbf{R}$ o υ, restringiendo los valores que estos parámetros podrían tomar, o como funciones de penalización o probabilidades a

priori [55], las cuales se añaden a las funciones objetivo (2.13), (2.14) y (2.17), y se usan para estimar $\mathbf{A}$, $\mathbf{s}$, $\mathbf{R}$ y υ con una regla MAP (máximo a posteriori).

- **Modelado y explotado de la información espacial.**

  Una forma de introducir información en la separación a ciegas de fuentes es dar cuenta del hecho de que los vectores de mezcla $\mathbf{a}_j(f)$ y las matrices de covarianza espacial $\mathbf{R}_j(f)$ no son independientes en frecuencia, si no que tienen ciertas dependencias, debido a las propiedades espaciales de las fuentes y de la sala donde se produce la grabación. Vamos a introducir algunas propiedades que se pueden explotar en este contexto. Cada modelo presentado incluye la información aportada por los modelos anteriores y añade alguna nueva.

  – **Localización espacial.**

    En campo abierto, los vectores de mezcla $\mathbf{a}_j(f)$ serían colineales con

    $$\mathbf{d}_j(f) = \left[\frac{1}{r_{1j}}e^{-2i\pi f r_{1j}/c}, \ldots, \frac{1}{r_{Ij}}e^{-2i\pi f r_{Ij}/c}\right]^T \tag{2.19}$$

    que es el vector de dirección que modela la atenuación del sonido y el retraso desde la fuente a los micrófonos, siendo c la velocidad del sonido y $\mathbf{r}_{ij}$ la distancia de la fuente j-ésima al micrófono i-ésimo. En condiciones reales de grabación, $\mathbf{a}_j(f)$ se desvía de $\mathbf{d}_j(f)$ debido a las reflexiones en los límites de la habitación, lo que incluye ecos y reverberación.

    Parra y Alvino [46] fueron los primeros en explotar la aproximación de $\mathbf{a}_j(f)$ a $\mathbf{d}_j(f)$, definiendo un término de penalización $\mathscr{P}(\mathbf{A}(f))$ sobre la matriz de mezcla. Se han sugerido diferentes términos de penalización, siendo uno de los más simples la distancia euclídea al cuadrado entre $\mathbf{a}_j(f)$ y $\mathbf{d}_j(f)$

    $$\mathscr{P}(\mathbf{a}_j(f)) = |\mathbf{a}_j(f) - \mathbf{d}_j(f)|_2^2. \tag{2.20}$$

    Sawada et al. [51] demostraron que, minimizar (2.20) con respecto a $\mathbf{r}_{ij}$ equivale a localizar la fuente a través de la técnica de las correlaciones cruzadas. Esto llevó a un enfoque iterativo conjunto para localización y separación de fuentes, donde las señales de dichas fuentes y las localizaciones de éstas se actualizan alternativamente.

  – **Anchura espacial.**

    Duong et al. [17] más tarde observaron que, la aproximación de banda estrecha (2.12) es inválida para fuentes reverberadas y/o difusas espacialmente: el sonido emitido por cada fuente, alcanza a los micrófonos por diferentes direcciones a la vez en cada frecuencia, en vez de haber una única dirección aparente $\mathbf{a}_j(f)$, de modo que los canales de $\mathbf{c}_j(n,f)$ son parcialmente incorrelados. La extensión de la distribución de las direcciones entrantes rige la anchura espacial percibida de la fuente en esa frecuencia. Se introdujo el concepto de matriz espacial de covarianza de rango completo $\mathbf{R}_j(f)$, que en comparación con la de rango unitario, considera no solo la localización espacial de las fuentes, sino que también tiene en cuenta su anchura.

    Asumiendo que las distancias de las fuentes a los micrófonos son conocidas pero que sus posiciones absolutas en la sala no lo son, la media de $\mathbf{R}_j(f)$ sobre estas localizaciones absolutas desconocidas es aproximadamente igual a [18]

    $$\mu_{R_j}(f) = \mathbf{d}_j(f)\mathbf{d}_j^H(f) + \sigma_{ech}^2\Omega(f). \tag{2.21}$$

    El primer término afecta al sonido directo, modelado por el vector direccional $\mathbf{d}_j(f)$ en (2.19), y el segundo término, a los ecos y la reverberación, modelado por la potencia de los ecos y la reverberación σ_{ech}^2 y por la matriz de covarianza de un campo sonoro isotrópico $\Omega(f)$.

  – **Ecos tempranos y reverberación.**

    Aunque el modelo de rango completo (2.1.2) mejorase considerablemente al de banda estrecha (2.12), sigue siendo una aproximación al proceso de mezcla real. La Figura 2.3 ilustra la forma de la respuesta impulsiva de una sala, $\mathbf{a}_{ij}(t)$, sobre el tiempo. En condiciones típicas de reverberación,

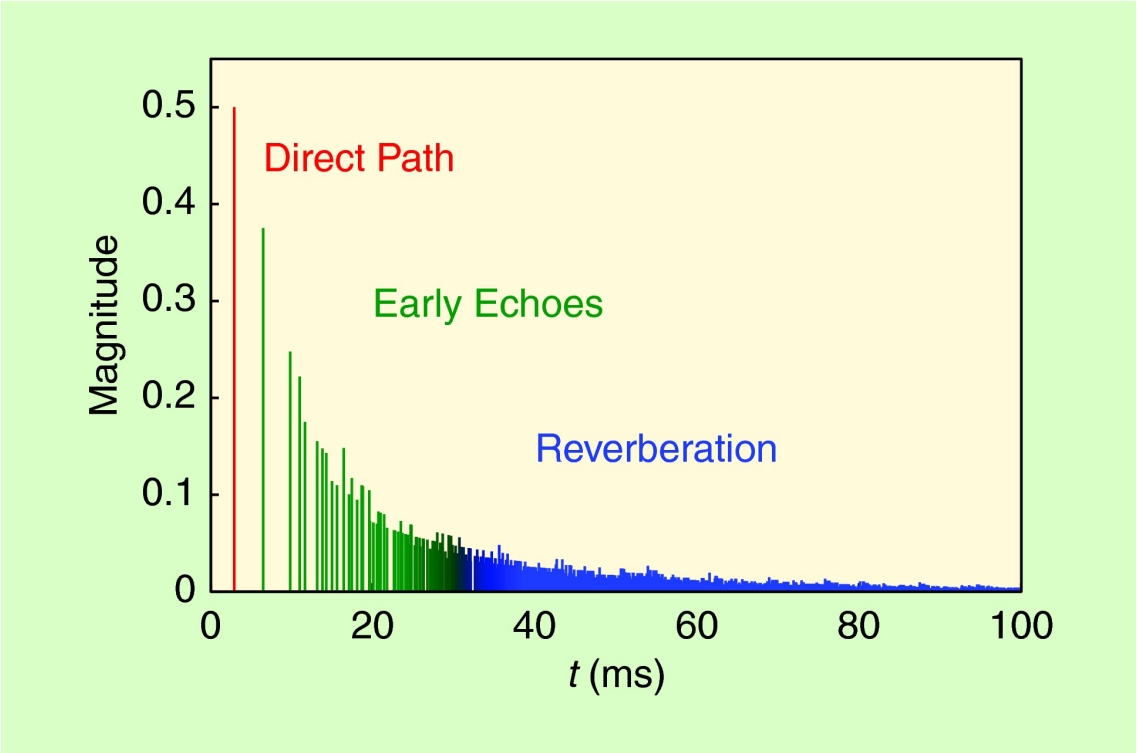

Figura 2.3 Ilustración esquemática de la magnitud de la respuesta impulsiva de una sala entre una fuente y un micrófono para un tiempo de reverberación $RT_{60} = 250ms$ [55].

estas repuestas tienen longitudes del orden de varios centenares de milisegundos, así que se extienden sobre numerosos intervalos de tiempo. Esto llevó a los investigadores a generalizar la Ecuación (2.1.2) en el caso mono canal como la convolución entre $\upsilon_j(n,f)$ y un filtro no negativo exponencial decreciente $\mathbf{q}_j(l,f)$, representando la potencia de $\mathbf{a}_j(t)$ para un retraso de l intervalos de tiempo [24]. Este modelo ha sido usado para procesos de reducción de reverberación de una única fuente, siendo RT_{60} conocido, y está empezando a usarse para problemas de BSS.

Kowalski et al. [33] fueron un paso más allá, al discutir sobre volver al dominio temporal para el modelado de los filtros de mezcla, mientras seguían explotando la dispersión de las fuentes en el dominio tiempo-frecuencia. Este estudio fue el punto de partida para numerosos estudios posteriores basados en definir funciones de penalización sobre los filtros de mezcla en el dominio temporal.

– **Acústica de sala llena.**

Últimamente, buena parte de los investigadores involucrados en la separación de fuentes de audio han propuesto parar de modelar las respuestas impulsivas entre fuentes individuales y micrófonos para estudiarlas entre todos los posibles pares de puntos de la sala, bajo la restricción de que el sistema de separación de fuente debe ser usado siempre en esa sala. Lo lógico es que las respuestas impulsivas de una sala abarquen una variedad, es decir, que un pequeño movimiento en la sala implica una pequeña desviación en la respuesta impulsiva, de modo que la medición de la respuesta impulsiva para algunos puntos debe ser suficiente para predecirla en otros puntos. Esto explica toda la información disponible posible, incluyendo el camino directo, los retardos y las amplitudes de los ecos tempranos, y la forma de la reverberación. *Asaei et al.* [1] consideraron cada punto de la sala como una fuente y limitaron la mayoría de las fuentes a estar inactivas por medio de una penalización de dispersión grupal. Más recientemente, *Deleforge et al.* [15] han intentado desarrollar una representación de menor dimensión mediante la incrustación lineal local probabilística. La última aproximación consiguió resultados considerablemente mejores para la separación de fuentes dadas cientos de medidas de la respuesta impulsiva de la sala, y

su extensión a escenarios prácticos, con menos mediciones, constituye un gran avance para la investigación en este campo.

- **Modelado y explotado de la información espectro-temporal.**

  Además de la información espacial, el espectro de las fuentes y su evolución temporal son el segundo suministrador principal de información para la separación de fuentes de audio. A continuación, se hace una revisión de propiedades complejas de $\mathbf{s}_j(n,f)$ y $\upsilon_j(n,f)$, que pueden ser usadas como guías en la separación, desde la persistencia local a las dependencias a largo plazo.

  – **Persistencia en tiempo-frecuencia.**

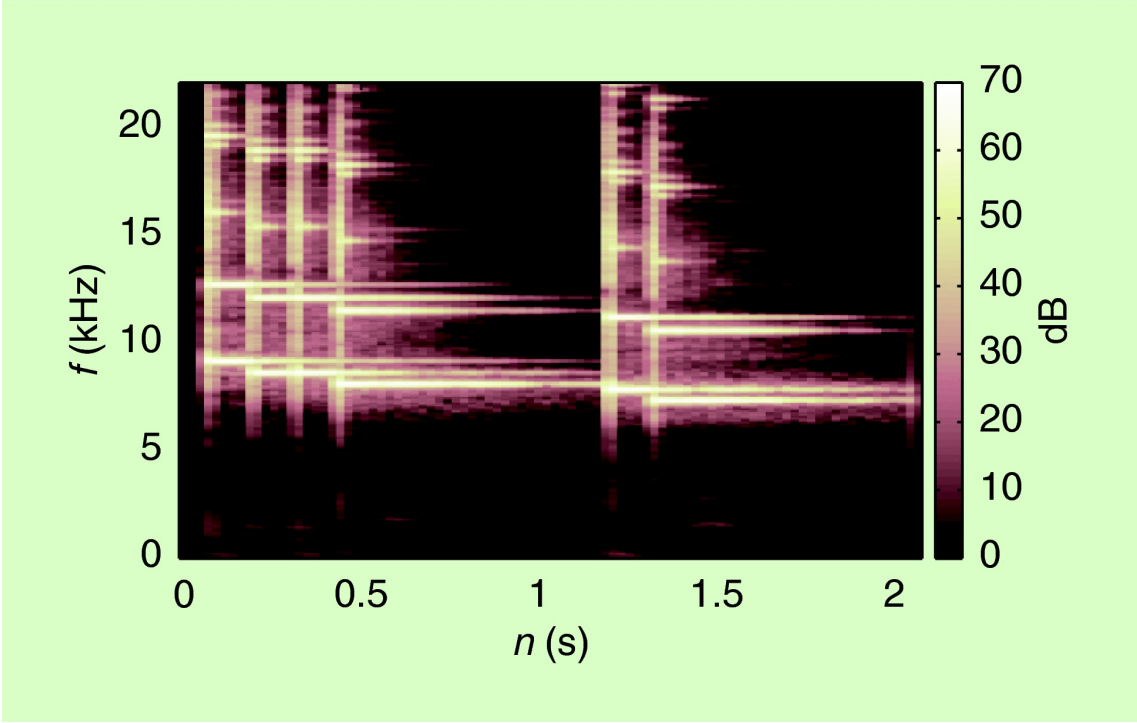

Figura 2.4 Espectrograma de una melodía tocada en un xilófono [55].

En señales de audio, los coeficientes significativos de la STFT no se distribuyen de forma aleatoria en el dominio tiempo-frecuencia, tienden a agruparse. Estos se ilustra en la Figura 2.4, donde aparecen las líneas verticales y horizontales, correspondientes a las partes transitorias y tonales de notas musicales respectivamente. En audios correspondientes al habla, aparecen estructuras similares y más complejas. Esta persistencia sobre el tiempo o frecuencia puede promoverse con el uso de dispersiones u otras penalizaciones estructuradas de dispersión en $\mathbf{s}_j(n,f)$ [34]. Por ejemplo, la norma $\ell_{1,2}$

$$\mathscr{P}(\mathbf{s}_j) = \sum_n \sqrt{\sum_f |\, \mathbf{s}_j(n,f)\,|^2} \tag{2.22}$$

que impone la dispersión sobre el tiempo, pero no sobre la frecuencia.

– **Espectro a corto plazo.**

Además de por la persistencia frecuencial, las fuentes de audio se caracterizan por su espectro a corto plazo, es decir, las dependencias entre $\upsilon_j(n,f)$ sobre todo el rango de frecuencias f. Una aproximación muy usada es la de representar el espectro a corto plazo de la fuente $\upsilon_j(n,f)$ como la suma de espectros bases no negativos $\mathbf{w}_{jk}(f)$, escalados por coeficientes no negativos variantes en el tiempo $\mathbf{h}_{jk}(n)$ [59], [42]:

$$\upsilon_j(n,f) = \sum_{k=1}^{K} \mathbf{w}_{jk}(f)\mathbf{h}_{jk}(n). \tag{2.23}$$

Este modelo ha sido aplicado tanto a espectros de magnitud como a espectros de potencia en el caso mono canal. En los últimos tiempos ha empezado a aplicarse a casos multi canal. Cada espectro base puede representar, por ejemplo, parte de un fonema o de una nota musical, como se ilustra en la Figura 2.5(a). Debido a su forma matricial equivalente $\mathbf{V}_j = \mathbf{W}_j\mathbf{H}_j$, este modelo es más conocido como *Factorización No Negativa de Matrices* (**NMF**), modelo en el que se basa nuestro trabajo y al que le dedicaremos una sección posteriormente. Considerando el hecho de que un solo fonema en una conversación, o unas pocas notas musicales en una pista de audio, pueden estar activas a la vez, la dispersión se hace cumplir reduciendo la suma a un solo componente k o añadiendo penalizaciones como la norma ℓ_1, $\mathscr{P}(\mathbf{H}_j) = \sum_{k,n} |\, \mathbf{h}_{jk}(n) \,|$ [59]. También se introdujeron las penalizaciones y prioridades para la dispersión por grupos de cara a favorecer la actividad simultánea de espectros base asociados al mismo fonema o nota, o para seleccionar al orador o al instrumento correcto entre una colección de espectros base entrenados en diferentes oradores e instrumentos [40].

– **Estructura espectral fina y envolvente espectral.**

Numerosas extensiones se aplicaron a NMF para mejorar las restricciones en las bases espectrales. La primera idea, es descomponer el espectro base con NMF como la suma de patrones espectrales de banda estrecha, $\mathbf{b}_{jkm}$, ponderados por coeficientes de la envolvente espectral $\mathbf{e}_{jkm}$:

$$\mathbf{w}_{jk}(f) = \sum_{m=1}^{M_k} \mathbf{b}_{jkm}(f)\mathbf{e}_{jkm}. \tag{2.24}$$

El espectro de banda estrecha, puede ser fijado para reforzar la armonía o la suavidad, las cuales son estructuras comunes en la mayoría de las fuentes de audio, y para adaptar los coeficientes de la envolvente espectral a la mezcla, los cuales son específicos de cada fuente. Estas estructuras son adecuadas para sonidos musicales tanto sostenidos como transitorios, como se aprecia en la Figura 2.5(b).

Otro refinamiento que cumple con la producción física de muchos sonidos naturales, es descomponer el espectro a corto plazo de la fuente mediante el modelo de excitación del filtro

$$\upsilon_j(n,f) = \upsilon_j^{ex}(n,f)\upsilon_j^{ft}(n,f), \tag{2.25}$$

donde $\upsilon_j^{ex}(n,f)$ y $\upsilon_j^{ft}(n,f)$ representan a la señal de excitación (por ejemplo, la glotis) y la respuesta del filtro (por ejemplo, el tracto vocal) y son modelados por NMF.

Ozerov et al. [44] recientemente propusieron un marco NMF multinivel exhaustivo que integre (2.23)-(2.25) mediante la multiplicación de hasta 8 matrices, cada una de ellas capaz de incorporar datos específicos o restricciones de forma flexible. Todas estas extensiones, pueden formalizarse de forma compacta como la factorización no negativa de tensores (NTF), una extensión de NMF para matrices multidimensionales que nos será muy útil en la parte experimental de este trabajo.

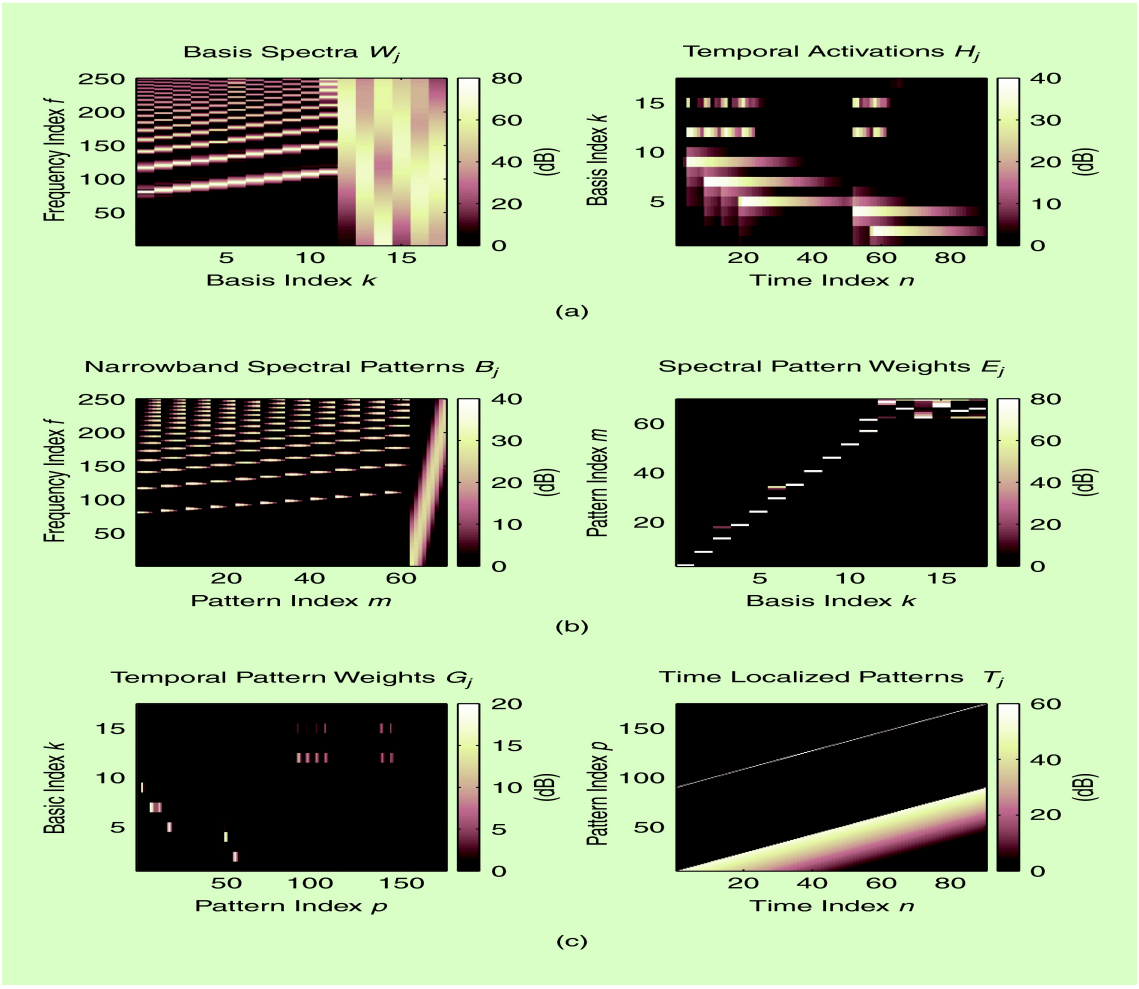

Figura 2.5 Descomposición NMF multinivel del espectrograma de la Figura 2.4.(a) Descomposición como producto entre el espectro base W_j y las plantillas de activación temporal H_j.(b)Descomposición de segundo nivel de W_j como el producto de los patrones espectrales armónicos y ruidosos de banda estrecha B_j y las envolventes espectrales asociadas E_j.(c) Descomposición de segundo nivel de H_j como producto de patrones localizados en el tiempo T_j activados en algún tiempo ponderado G_j [55].

– **Evolución temporal.**

Los modelos mencionados anteriormente no modelan directamente la evolución temporal del espectro. En una escala corta de tiempo, *Virtanen* [59] forzó la continuidad de los coeficientes de activación NMF añadiendo la función de penalización $\mathscr{P}(\mathbf{H}_j) = \sum_n \mid \mathbf{h}_{jk}(n+1) - \mathbf{h}_{jk}(n) \mid^2$ mientras que *Ozerov et al.* [44] lo modelaron de forma similar a (2.24), como el producto de patrones localizados en el tiempo y envolventes temporalmente escasas, como se muestra en la Figura 2.5(c).

En una escala de tiempo mediana, *Smaragdis* [52] generalizó (2.23) en el modelo NMF convolutivo

$$v_j(n,f) = \sum_{k=1}^{K} \sum_{l} \omega_{jk}(l,f)\mathbf{h}_{jk}(n-l), \tag{2.26}$$

donde los elementos base, $\mathbf{w}_{jk}(l,f)$, son ahora parches espectro-temporales en vez de espectros de un solo cuadro, codificando así de forma explícita la evolución temporal de los eventos sonoros en cada frecuencia.

Otro avance importante, ha sido el interés de muchos investigadores por explotar la información codificada mediante redundancia y patrones repetitivos en escalas de tiempo muy largas, para optimizar así el uso de la información disponible sobre la duración total de la señal. *Huang et al.* [27], usaron el Análisis Robusto de Componentes Principales (RCPA), el cual descompone un espectrograma de entrada como la suma de una matriz de rango bajo y una matriz dispersa, para separar fuentes de batería y melodía, de fuentes de acompañamiento tonal repetitivo. La búsqueda de patrones repetitivos en la música también ha sido explotado por *Rafii et al.* [49] mediante la identificación de segmentos repetidos (de un máximo de 40s), modelando y extrayendo a través de un enmascarado en tiempo-frecuencia.

2.2 Factorización No Negativa de Matrices (NMF) y Factorización No Negativa de Tensores (NTF)

El método desarrollado por [53] utiliza NTF, es por eso que en esta sección se va a desarrollar la técnica usada, tomando como referente para todo la sección el libro [13].

2.2.1 Introducción

La Factorización No Negativa de Matrices (Non-Negative Matrix Factorization, NMF), consiste en la descomposición de una matriz como producto de dos o más matrices. La única restricción que exige este método es que todos los coeficientes de las matrices han de ser positivos.

Las primeras referencias que se tienen sobre NMF son de *Paatero y Tapper* en unos trabajos publicados en 1991 [45], donde se expone el método como una variante de la Factorización Positiva de Matrices (PMF), aunque fue con los trabajos de *Lee y Seung* publicados en *Nature and NIPS* [37] [36] cuando ganó popularidad, ya que éstos aportaron los primeros algoritmos de aplicación. En la actualidad, NMF es uno de los métodos más usados en BSS.

En este problema se ha usado la Factorización No Negativa de Tensores (NTF), método análogo a NMF aplicado a tensores, entendiendo los tensores como matrices de N dimensiones o conjuntos de datos (datasets) indexados por N índices, donde N puede tomar valores mayores que 2 [19]. Para N=1, un tensor equivale a un escalar y para N=2 a una matriz, en nuestro trabajo usaremos tensores de N=4.

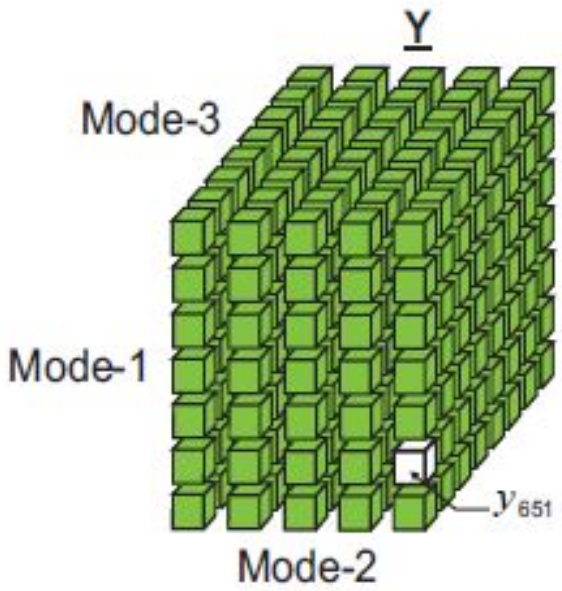

Figura 2.6 Tensor de N=3 [13].

La principal diferencia de NMF respecto a otros métodos de factorización, es la no negatividad de sus coeficientes, la cual es muy importante en la percepción. Muchos datos del mundo real son no negativos y las componentes ocultas solo tienen significado físico cuando son positivas. Esto ocurre en varios campos como el tratamiento de imagen y vídeo, economía y por supuesto en el que nos ocupa, el tratamiento de señales de audio. En este campo, la no negatividad cobra una gran importancia, ya que suele realizarse la separación de audio en el dominio tiempo-frecuencia, usando generalmente la magnitud de las componentes transformadas.

NMF es un modelo aditivo, en el que un valor cero representa la ausencia de componentes de la magnitud con la que se esté tratando y un número positivo representa la presencia de alguna componente, lo que permite que cada una de las partes que conforman la suma pueda ser considerada como parte de los datos originales. Gracias a esto, podemos mantener un buen equilibrio entre la interpretabilidad de los datos y la fidelidad estadística de los mismos, hecho que hace al método óptimo para nuestro trabajo.

De este tipo de factorización existen varias versiones, podemos hablar de NMF simétrica, convolutiva o multicapa entre otras. Estas diferentes versiones permiten simplificar los modelos en diferentes casos. En nuestro trabajo nos centraremos en el modelo básico, que es el más común y en NMF de Alta Resolución.

2.2.2 Modelo NMF básico

El problema básico de NMF se puede expresar de la siguiente manera: dada una matriz de coeficientes no negativos $\mathbf{Y} \in \mathbb{R}_+^{J\times T}$ ($y_u \geq 0$ o equivalentemente $\mathbf{Y} \geq 0$) y un rango reducido J ($J \leq \text{mín}(I,T)$), el objetivo es encontrar dos matrices no negativas $\mathbf{A} = [\mathbf{a}_1,\mathbf{a}_2,\ldots,\mathbf{a}_J] \in \mathbb{R}_+^{I\times J}$ y $\mathbf{X} = \mathbf{B}^T = [\mathbf{b}_1,\mathbf{b}_2,\ldots,\mathbf{b}_J]^T \in \mathbb{R}_+^{J\times T}$ tales que factoricen $\mathbf{Y}$ lo mejor posible, eso es:

$$\mathbf{Y} = \mathbf{AX} + \mathbf{E} = \mathbf{AB}^T + \mathbf{E} \tag{2.27}$$

donde la matriz $\mathbf{E} \in \mathbb{R}^{I\times T}$ representa el error aproximado en la descomposición. Las matrices $\mathbf{A}$ y $\mathbf{X}$ pueden tener diferentes sentidos físicos, dependiendo de la aplicación. En los problemas de BSS, $\mathbf{A}$ representa la matriz de mezcla y $\mathbf{X}$ las señales fuente.

En NMF estándar, solo asumimos la no negatividad de las matrices $\mathbf{A}$ y $\mathbf{X}$. Al contrario que en los métodos para BSS basados en el Análisis de Componentes Independientes (ICA), aquí no se asume la independencia de las fuentes, en cambio, se introducen otras suposiciones y restricciones para $\mathbf{A}$ y/o $\mathbf{X}$ posteriormente. Esta simetría en las suposiciones, conduce a una simetría en la factorización: podríamos simplemente escribir $\mathbf{Y}^T \approx \mathbf{X}^T\mathbf{A}^T$, esto hace que a menudo el significado de "fuente" y "mezcla" en NMF sea algo arbitrario.

El modelo NMF también puede ser representado como una forma especial del modelo bilineal, donde los vectores son no negativos (ver Figura 2.7):

$$\mathbf{Y} = \sum_{j=1}^{J} \mathbf{a}_j \circ \mathbf{b}_j + \mathbf{E} = \sum_{j=1}^{J} \mathbf{a}_j\mathbf{b}_j^T + \mathbf{E} \tag{2.28}$$

donde el símbolo $\circ$ representa el producto externo de dos vectores. Por lo tanto, podemos construir una representación aproximada de la matriz de datos no negativos $\mathbf{Y}$, como una suma de matrices no negativas de rango unidad $\mathbf{a}_j\mathbf{b}_j^T$. El caso en el que esta descomposición sea exacta ($\mathbf{E} = 0$), se llama Factorización No Negativa de Rango (Nonnegative Rank Factorization, NRF), este caso en la realidad es muy complejo de conseguir, por lo que en este trabajo se considera la descomposición como una aproximación a la naturaleza, pero no exacta.

Aunque NMF se puede aplicar a los problemas de BSS para fuentes y matrices de mezcla no negativas, su aplicación no está limitada a la BSS, de hecho, puede ser usada en diversas aplicaciones. En varias de estas otras aplicaciones se requieren restricciones adicionales para los elementos de las matrices $\mathbf{A}$ y/o $\mathbf{X}$, como suavidad, dispersión, simetría y ortogonalidad.

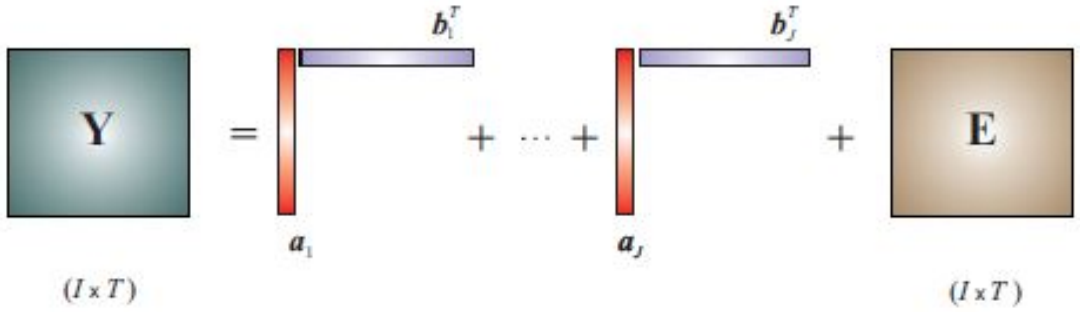

Figura 2.7 Modelo NMF bilineal. La aproximación de la matriz de datos no negativos $\mathbf{Y} \in \mathbb{R}_+^{I \times T}$ se representa con una suma o una combinación lineal de matrices no negativas de rango unidad $\mathbf{Y}^{(j)} = \mathbf{a}_j \circ \mathbf{b}_j = \mathbf{a}_j \mathbf{b}_j^T \in \mathbb{R}_+^{I \times T}$ [13].

2.2.3 Casos particulares de NMF

Como se ha expuesto en el inicio de este capítulo, para este tipo de factorización existen varios casos particulares derivados del modelo básico, aunque no se han usado en este trabajo se van a exponer brevemente para tener una idea más amplia del alcance de esta factorización.

NMF simétrica

Para el caso particular en el que $\mathbf{A} = \mathbf{B} \in \mathbb{R}_+^{I \times J}$, la descomposición se denomina NMF simétrica, y puede expresarse como:

$$\mathbf{Y} = \mathbf{A}\mathbf{A}^T + \mathbf{E} \tag{2.29}$$

Si existe la simetría exacta (cuando $\mathbf{E} = 0$), se dice que la matriz no negativa $\mathbf{Y} \in \mathbb{R}_+^{I \times I}$ es *completamente positiva* (CP).

NMF semi-ortogonal

Se define igual que el modelo básico:

$$\mathbf{Y} = \mathbf{A}\mathbf{X} + \mathbf{E} = \mathbf{A}\mathbf{B}^T + \mathbf{E}, \tag{2.30}$$

la diferencia radica en que, además de la restricción de no negatividad de las matrices $\mathbf{A}$ y $\mathbf{X}$, se añade la de ortogonalidad: $\mathbf{A}^T\mathbf{A} = \mathbf{I}_j$ o $\mathbf{X}\mathbf{X}^T = \mathbf{I}_j$.

Semi-NMF

En algunas aplicaciones, los datos de entrada observados no tienen signo: $\mathbf{Y} = \mathbf{Y}_\pm \in \mathbb{R}^{I \times T}$. Esto nos permite relajar las restricciones con respecto a la no negatividad de las matrices. Así, Semi-NMF se puede expresar como:

$$\mathbf{Y}_\pm = \mathbf{A}_\pm \mathbf{X}_+ + \mathbf{E}, \qquad or \qquad \mathbf{Y}_\pm = \mathbf{A}_+ \mathbf{X}_\pm + \mathbf{E}, \tag{2.31}$$

Tri-NMF

También conocida como *NMF de tres factores*. Es un caso particular de NMF multicapa, en el que entra en juego una nueva matriz, quedando el modelo de la siguiente forma:

$$\mathbf{Y} = \mathbf{A}\mathbf{S}\mathbf{X} + \mathbf{E}, \tag{2.32}$$

donde las restricciones de no negatividad pueden ser impuestas a todas o solo a las matrices de factorización elegidas: $\mathbf{A} \in \mathbb{R}^{I \times J}$, $\mathbf{S} \in \mathbb{R}^{J \times R}$, y/o $\mathbf{X} \in \mathbb{R}^{R \times T}$. Si no se añaden restricciones adicionales en la factorización, este modelo se puede reducir al estándar con la transformación $\mathbf{A} \leftarrow \mathbf{A}\mathbf{S}$ o $\mathbf{X} \leftarrow \mathbf{S}\mathbf{X}$. Sin embargo, Tri-NMF no es equivalente al modelo básico si aplicamos restricciones o condiciones especiales, así aparecen varios modelos como: Tri-NMF Ortogonal, Tri-NMF No Suave, Filtrado NMF o la Descomposición CGR/CUR.

NMF con offset

El objetivo es eliminar el valor de referencia o el nivel de continua de la matriz $\mathbf{Y}$, usando un modelo NMF ligeramente modificado:

$$\mathbf{Y} = \mathbf{A}\mathbf{X} + \mathbf{a}_0\mathbf{l}^T + \mathbf{E}, \tag{2.33}$$

donde $\mathbf{l} \in \mathbb{R}^T$ es un vector todo unos y $\mathbf{a}_0 \in \mathbb{R}_+^I$ es un vector escogido para que la matriz $\mathbf{X}$ tenga la tierra a cero. El término $\mathbf{Y}_0 = \mathbf{a}_0\mathbf{l}^T$ denota el offset, que junto a la restricción de no negatividad, a menudo asegura la poca dispersión de las matrices factorizadas. El papel principal de este término es absorber los valores constantes de la matriz de datos.

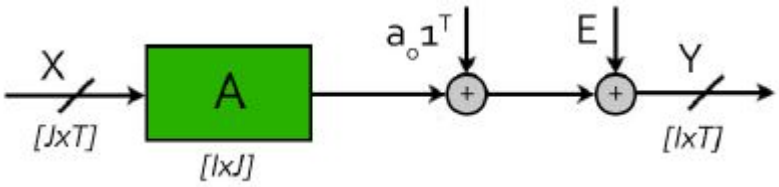

Figura 2.8 Esquema NMF con offset [13].

NMF multicapa

En este caso, la matriz $\mathbf{A}$ se remplaza por un conjunto de matrices en cascada (capas). El modelo se describe como (ver Figura 2.9):

$$\mathbf{Y} = \mathbf{A}^{(1)}\mathbf{A}^{(2)}\cdots\mathbf{A}^{(L)}\mathbf{X} + \mathbf{E}, \tag{2.34}$$

Como el modelo es lineal, todas las matrices pueden ser fusionadas en una sola matriz $\mathbf{A}$, si no se han impuesto restricciones especiales a las matrices que conforman las capas. Este modelo se puede utilizar para mejorar considerablemente el rendimiento del modelo NMF estándar gracias a la estructura distribuida en capas y al alivio del problema de los mínimos locales.

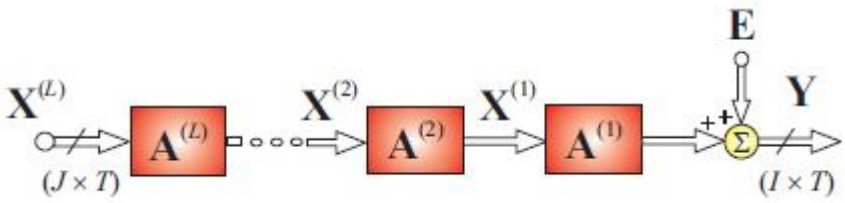

Figura 2.9 Esquema NMF multicapa [13].

NMF simultánea

En NMF Simultánea se tienen dos o más matrices de entrada de datos vinculadas (llamadas $\mathbf{Y}_1$ e $\mathbf{Y}_2$) y el objetivo es descomponerlas en matrices de factorización no negativas de forma que una de las matrices de factorización sea común a ambas, por ejemplo:

$$\begin{aligned} \mathbf{Y}_1 &= \mathbf{A}_1\mathbf{X} + \mathbf{E}_1, \\ \mathbf{Y}_2 &= \mathbf{A}_2\mathbf{X} + \mathbf{E}_2, \end{aligned} \tag{2.35}$$

NMF proyectiva

Un modelo NMF Proyectivo puede formularse como la estimación de una matriz dispersa y no negativa $\mathbf{W} \in \mathbb{R}_+^{I\times J}$ que satisfaga la ecuación matricial:

$$\mathbf{Y} = \mathbf{W}\mathbf{W}^T\mathbf{Y} + \mathbf{E}, \tag{2.36}$$

En una forma general no simétrica, NMF proyectiva implica la estimación de 2 matrices no negativas: $\mathbf{A} \in \mathbb{R}_+^{I\times J}$ y $\mathbf{B} \in \mathbb{R}_+^{I\times J}$ en el modelo (ver Figura 2.10):

$$Y = \mathbf{A}\mathbf{B}^T\mathbf{Y} + \mathbf{E} \tag{2.37}$$

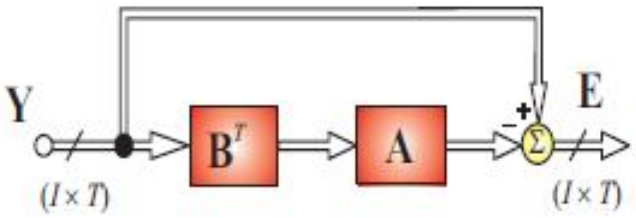

Figura 2.10 Esquema NMF Proyectiva [13].

NMF convexa

En NMF convexa se asume que los vectores base $\mathbf{A} = [\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_J]$ tienen como restricción ser combinaciones convexas de la matriz de datos de entrada $Y = [\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_T]$. Es decir:

$$\mathbf{a}_j = \sum_{t=1}^{T} w_{tj} y_t = \mathbf{Y} w_j \qquad o \qquad \mathbf{A} = \mathbf{YW}, \tag{2.38}$$

donde $\mathbf{W} \in \mathbb{R}_+^{T \times J}$ y $\mathbf{X} = \mathbf{B}^T \in \mathbb{R}_+^{J \times T}$. El modelo NMF Convexo puede ser escrito de forma matricial como:

$$\mathbf{Y} = \mathbf{YWX} + \mathbf{E} \tag{2.39}$$

aplicando el operador de transposición obtenemos:

$$\mathbf{Y}^T = \mathbf{X}^T \mathbf{W}^T \mathbf{Y}^T + \mathbf{E}^T \tag{2.40}$$

En la Figura 2.11 se puede apreciar que, NMF convexa se puede representar de una forma similar a NMF Proyectiva.

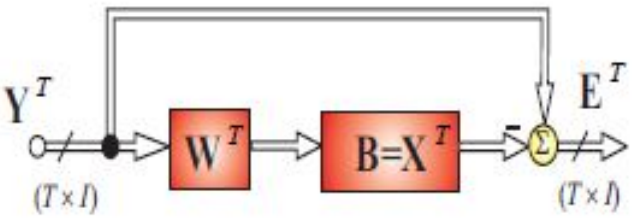

Figura 2.11 Esquema NMF Convexa [13].

Kernel NMF

Considere un mapeo $y_t \rightarrow \phi(y_t)$ o $\mathbf{Y} \rightarrow \phi(\mathbf{Y}) = [\phi(y_1), \phi(y_2), \ldots, \phi(y_T)]$, así Kernel NMF puede definirse como:

$$\phi(\mathbf{Y}) \cong \phi(\mathbf{Y})\mathbf{W}\mathbf{B}^T. \tag{2.41}$$

NMF convolutiva

Este caso es una generalización de NMF básica, donde se trabaja con versiones de la matriz $\mathbf{X}$ desplazadas horizontalmente. Matemáticamente, se puede expresar el modelo como:

$$\mathbf{Y} = \sum_{p=0}^{P-1} \mathbf{A}_p \overset{p\rightarrow}{\mathbf{X}} + \mathbf{E}, \tag{2.42}$$

donde $\mathbf{X} = \overset{0\rightarrow}{\mathbf{X}}$, representa la matriz de fuentes primaria, y los términos $\overset{p\rightarrow}{\mathbf{X}}$ representan los vectores de la matriz $\mathbf{X}$ desplazados p columnas. Las columnas desplazadas hacia fuera son fijadas a cero, tal como puede

verse en el siguiente ejemplo:

$$\mathbf{X} = \begin{bmatrix} 1 & 3 & 5 \\ 2 & 4 & 6 \end{bmatrix} \quad \overset{1\rightarrow}{\mathbf{X}} = \begin{bmatrix} 0 & 1 & 3 \\ 0 & 2 & 4 \end{bmatrix} \quad \overset{2\rightarrow}{\mathbf{X}} = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 2 \end{bmatrix} \quad \overset{1\leftarrow}{\mathbf{X}} = \begin{bmatrix} 3 & 5 & 0 \\ 4 & 6 & 0 \end{bmatrix} \tag{2.43}$$

En la Figura 2.12 queda reflejado este modelo donde el operador $\mathbf{S}_p = \mathbf{T}_1$ denota el desplazamiento horizontal.

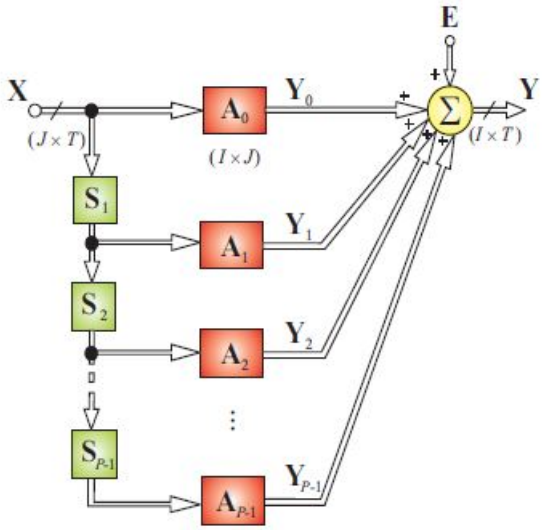

Figura 2.12 Esquema NMF Convolutiva [13].

NMF superpuesta

Se trata de una extensión del caso convolutivo, mientras que en éste se realiza un desplazamiento horizontal de las columnas, en el caso de NMF superpuesta, se realizan diferentes transformaciones, como desplazamientos verticales, muy útiles por ejemplo, a la hora de trabajar con espectrogramas.

La expresión matemática de este modelo varía en función de las transformaciones realizadas sobre las filas y columnas de la matriz $\mathbf{X}$, por ejemplo, podría expresarse como:

$$\mathbf{Y} \approx \sum_{p=0}^{P} (\overset{\rightarrow p}{\mathbf{X}})^T \mathbf{A}_p^T = \sum_{p=0}^{P} (\mathbf{X}\mathbf{T}_p)^T \mathbf{A}_p^T = \sum_{p=0}^{P} \mathbf{T}_p^T \mathbf{X}^T \mathbf{A}_p^T, \tag{2.44}$$

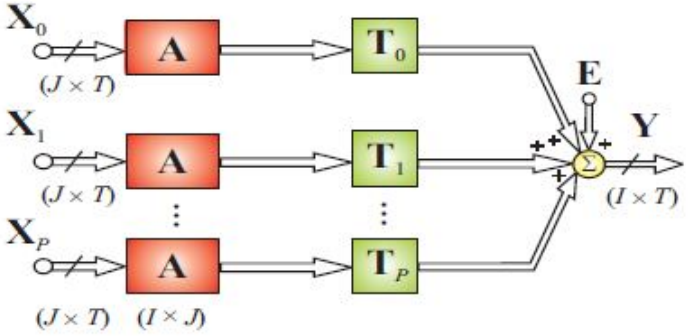

Figura 2.13 Esquema NMF Superpuesta [13].

2.2.4 NMF de alta resolución (High Resolution NMF, HR-NMF)

Debido a que éste es el modelo que más se asemeja al que se desarrolla en [53], se va a introducir en esta sección. Este modelo fue presentado en 2011 por *Roland Badeau* en [3], se generalizó para mezclas multicanal en [5] y se demostró que proporciona un rendimiento considerablemente mejor para la separación de fuentes que los modelos anteriores en [48]. Aunque algunas aproximaciones variacionales fueron introducidas en [4] para reducir su complejidad, estos algoritmos son, a menudo, muy exigentes para aplicaciones prácticas.

Según el trabajo de *Roland Badeau y A.Dremeau* [3], HR-NMF es un modelo que permite superar las limitaciones de resolución espectral que tiene el modelo NMF, teniendo en cuenta tanto la fase, como las correlaciones locales en cada banda de frecuencia. Este modelo, se estima implementando de forma recursiva un algoritmo EM [16], que se aplica de forma satisfactoria a los problemas de separación de fuentes.

A continuación, vamos a introducir el modelo presentado en [3]. El modelo de mezcla $x(f,t)$, se define para todas las frecuencias $1 \leq f \leq F$ y tiempos $1 \leq t \leq T$ como la suma de K componentes ocultas $c_k(f,t)$ más un ruido blanco $n(f,t) \sim \mathcal{N}(0,\sigma^2)$:

$$x(f,t) = n(f,t) + \sum_{k=1}^{K} c_k(f,t) \tag{2.45}$$

donde

- $c_k(f,t) = \sum_{p=1}^{P(k,f)} a(p,k,f)c_k(f,t-p) + b_k(f,t)$ se obtiene filtrando de forma autorregresiva una señal no estacionaria $b_k(f,t)$ (y $P(k,f) \in \mathbb{N}$ tal que $a(P(k,f),k,f) \neq 0$),
- $b_k(f,t) \sim \mathcal{N}(0,\upsilon_k(f,t))$ donde $\upsilon_k(f,t)$ se define como

$$\upsilon_k(f,t) = w(k,f)h(h,t), \tag{2.46}$$

  con $w(k,f) \geq 0$ y $h(k,t) \geq 0$,
- Los procesos n y $b_1 \dots b_K$ son mutuamente independientes.

Dado que $\mathcal{N}$ denota tanto la distribución normal real como la circular compleja, el modelo (2.45) puede tomar valores reales o complejos. Además, para instantes anteriores al inicial se asume $c_k(f,t) \sim \mathcal{N}(0,1)$ y no se dispone de las observaciones $x(f,t)$. Los parámetros a estimar son σ^2, $a(p,k,f)$, $w(k,f)$ y $h(k,t)$.

Este modelo en el dominio tiempo-frecuencia, ha servido para generalizar algunos modelos muy usados en varios sectores del procesado de señal:

- Si $\sigma^2 = 0$ y $\forall k,f,P(k,f) = 0$, el modelo (2.45) se convierte en $x(f,t) = \sum_{k=1}^{K} b_k(f,t)$, así $x(f,t) \sim \mathcal{N}(0,\hat{\mathbf{V}}_{ft})$, donde $\hat{\mathbf{V}}$ se define por NMF como $\hat{\mathbf{V}} = \mathbf{WH}$ con $W_{fk} = w(k,f)$ y $H_{kt} = h(k,t)$. La estimación de máxima verosimilitud de $\mathbf{W}$ y $\mathbf{H}$ es entonces equivalente a la minimización de la divergencia de Itakura-Saito entre la matriz modelo $\hat{\mathbf{V}}$ y el espectrograma $\mathbf{V}$ (donde $V_{ft} = \mid x(f,t) \mid^2$), por ello este modelo toma el nombre de IS-NMF [20].
- Para valores conocidos de k y f, si $\forall t$, $h(k,t) = 1$, entonces $c_k(f,t)$ es un proceso autorregresivo de orden $P(k,f)$.
- Para valores conocidos de k y f, si $P(k,f) \geq 1$ y $\forall t \geq P(k,f)+1$, $h(k,t) = 0$, entonces $c_k(f,t)$ se puede escribir como $c_k(f,t) = \sum_{p=1}^{P(k,f)} \alpha_p z_p^t$ donde $z_1 \dots z_{P(k,f)}$ son las raíces del polinomio $z^{P(k,f)} - \sum_{p=1}^{P(k,f)} a(p,k,f) z^{P(k,f)-p}$. Esto corresponde al Modelo Exponencial Sinusoidal (ESM), frecuentemente usado en análisis espectral HR de series temporales [2].

Por todas estas razones, nos referimos al modelo (2.45) como HR-NMF.

2.2.5 Estimación de los parámetros NMF

Estimación basada en la medida

Para estimar las matrices de factorización $\mathbf{A}$ y $\mathbf{X}$ en el estándar de NMF, es necesario considerar alguna medida de similitud para cuantificar la diferencia entre la matriz de datos $\mathbf{Y}$ y su aproximación NMF $\hat{\mathbf{Y}} = \mathbf{AX}$. La elección de la medida de similitud (también llamada distancia o divergencia), depende mayormente de la distribución de probabilidad de las señales estimadas o de las componentes y la estructura de los datos o del ruido.

Una vez elegida la distancia, la función de coste será la función dada por dicha distancia, y el objetivo, obtener un algoritmo que permita minimizar dicho coste. A continuación, se expondrán algunas de las distancias más generalizadas, incluyendo un sencillo algoritmo de aplicación.

Norma de Frobenius

Esta norma, que toma nombre del matemático alemán *Ferdinand Georg Frobenius* es en la que se basa la medida más simple y comúnmente usada:

$$D_F(\mathbf{Y} \parallel \mathbf{AX}) = \frac{1}{2} \parallel \mathbf{Y} - \mathbf{AX} \parallel_F^2 \tag{2.47}$$

Se debe resaltar que la función de costes anterior es convexa con respecto, tanto a los elementos de la matriz $\mathbf{A}$, como a los de la matriz $\mathbf{X}$, no para ambos, si intentamos optimizar los dos se pierde la convexidad.

La minimización de la función de costes dada por la norma de Frobenius, da lugar al algoritmo de minimización de los mínimos cuadrados (en inglés, *"Alternating Least Squares"*, ALS), que es uno de los métodos de optimización más conocidos. A continuación, se describe este algoritmo con unos sencillos pasos:

1. Inicializar $\mathbf{A}$ de forma aleatoria o usando una estrategia determinista específica.
2. Estimar $\mathbf{X}$ de la ecuación matricial $\mathbf{A}^T\mathbf{AX} = \mathbf{A}^T\mathbf{Y}$ resolviendo

$$\min_X D_F(\mathbf{Y} \parallel \mathbf{AX}) = \frac{1}{2} \parallel \mathbf{Y} - \mathbf{AX} \parallel_F^2 \qquad \text{siendo} \quad \mathbf{A} \quad \text{fija.} \tag{2.48}$$

3. Fijar a cero, o a algún valor positivo pequeño, todos los elementos negativos de $\mathbf{X}$.
4. Estimar $\mathbf{A}$ de la ecuación matricial $\mathbf{XX}^T\mathbf{A}^T = \mathbf{XY}^T$ resolviendo

$$\min_A D_F(\mathbf{Y} \parallel \mathbf{AX}) = \frac{1}{2} \parallel \mathbf{Y}^T - \mathbf{X}^T\mathbf{A}^T \parallel_F^2 \qquad \text{siendo} \quad \mathbf{X} \quad \text{fija.} \tag{2.49}$$

5. Fijar a cero. o a algún valor positivo pequeño, ε, todos los elementos negativos de $\mathbf{A}$.

Observando estos pasos, se hace evidente la sencillez de este algoritmo, que podemos resumir mediante las siguientes ecuaciones:

$$\begin{aligned} \mathbf{X} &\rightarrow \{\varepsilon, (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{Y}\} = [\mathbf{A}^\dagger\mathbf{Y}]_+, \\ \mathbf{A} &\rightarrow \{\varepsilon, \mathbf{YX}^T(\mathbf{XX}^T)^{-1}\} = [\mathbf{YX}^\dagger]_+, \end{aligned} \tag{2.50}$$

donde $\mathbf{A}^\dagger$ es la inversa de Moore-Penrose de $\mathbf{A}$ y ε es una constante pequeña (típicamente 10^{-16}), que se usa para forzar a las entradas a ser positivas. Varias restricciones adicionales pueden imponerse sobre $\mathbf{A}$ y $\mathbf{X}$
.

Por último, decir que se ha considerado oportuno exponer aquí el algoritmo ALS debido a que es tomado como un enfoque básico en algunos métodos actuales, siendo utilizado frecuentemente en inicializaciones previas a la aplicación de otros algoritmos más complejos. Tiene la ventaja de que su implementación es bastante sencilla, aunque no garantiza la convergencia hacia mínimos globales y sus soluciones no son muy precisas.

Divergencia de Kullback-Leibler (KL)

Otra función de costes popular en NMF es la divergencia Kullback-Leibler, también llamada divergencia de la información o divergencia-I, es un caso especial de la llamada divergencia de Bregman y se define como:

$$D_{KL}(\mathbf{Y} \parallel \mathbf{AX}) = \sum_{it} \left(y_{it} \ln \frac{y_{it}}{[\mathbf{AX}]_{it}} - y_{it} + [\mathbf{AX}]_{it} \right). \tag{2.51}$$

Esta medida fue introducida en 1951 por *Solomon Kullback* y *Richard Leibler*, como una divergencia dirigida entre dos distribuciones [35]. Actualmente es muy usada en estadística y está muy relacionada con el método de ajuste de distribuciones por máxima verosimilitud.

Divergencia de Itakura-Saito (IS)

La divergencia IS, al igual que la divergencia KL, es una extensión de la divergencia de Bregman, muy usada en la estimación de los parámetros de NMF y que puede definirse de la siguiente forma:

$$D_{IS}(\mathbf{Y} \parallel \mathbf{AX}) = \sum_{it} \left(\ln \frac{[\mathbf{AX}]_{it}}{y_{it}} + \frac{y_{it}}{[\mathbf{AX}]_{it}} - 1 \right). \tag{2.52}$$

Fue propuesta por *Fumitada Itakura* y *Shuzo Saito*, cuando trabajaban en la compañía NTT (Nipppon Telegraph and Telephone) [29]. En la actualidad es una de las medidas más usadas en las técnicas de separación ciega de fuentes de audio.

2.2.6 Otros aspectos a considerar en NMF

Inicialización de parámetros

La solución y la convergencia dada por los algoritmos NMF, normalmente dependen mucho de las condiciones iniciales, es decir, sus valores iniciales supuestos, especialmente en un contexto multivariable. Por ello, es importante tener formas eficientes y consistentes de inicializar las matrices **A** y/o **X**. En otras palabras, la eficiencia de la mayoría de las estrategias NMF se ve claramente afectada por la selección de las matrices iniciales [13]. Inicializaciones pobres nos llevan a convergencias lentas y, en algunos casos, incluso a soluciones incorrectas o irrelevantes. Por otro lado, una determinada inicialización no se comporta igual para distintas matrices de entrada de datos, mientras que para unas puede aportar buenos resultados para otras puede resultar pobre, por ejemplo, este problema puede volverse bastante complejo cuando se trata con matrices factorizadas a las que se han impuesto ciertas limitaciones. Resulta útil, para evaluar la eficacia de la estrategia de inicialización y del propio algoritmo, realizar análisis de incertidumbre como simulaciones de Monte Carlo.

A continuación, a modo de ejemplo, se presentan una serie de pasos, que proporcionarían una estrategia de inicialización robusta:

1. Generar de forma iterativa un número R de matrices iniciales **A** y **X** (normalmente con 10-20 iteraciones es suficiente), mediante una inicialización aleatoria o cualquier algoritmo sencillo como el ALS.
2. Ejecutar algún algoritmo NMF específico para cada par de matrices iniciales generalizadas y con un número fijado de iteraciones (típicamente 10-20). Como resultado, se obtienen R estimaciones de las matrices $\mathbf{A}^{(r)}$ y $\mathbf{X}^{r}$.
3. Seleccionar de entre las estimaciones anteriores, aquella pareja que proporcione el menor valor para la función de coste, denotadas como $\mathbf{A}^{(r_{min})}$ y $\mathbf{X}^{(r_{min})}$.

Criterios de parada

Hay numerosos criterios de parada para los algoritmos iterativos usados en NMF, a continuación vamos a exponer algunos:

- Que la función de costes alcance el valor cero (o cercano a cero) o un valor por debajo de un umbral dado ε, por ejemplo,

$$D_F^{(k)}(\mathbf{Y} \parallel \hat{\mathbf{Y}}^{(k)}) = \parallel \hat{\mathbf{Y}}^{(k)} - \hat{\mathbf{Y}}^{(k+1)} \parallel_F^2 \leq \varepsilon, \tag{2.53}$$

  o

$$\frac{\mid D_F^{(k)} - D_F^{(k-1)}}{D_F^{(k)}} \leq \varepsilon. \tag{2.54}$$

- Que haya cambios insignificantes o que no haya cambios en las iteraciones sucesivas de las matrices $\mathbf{A}$ y $\mathbf{X}$.
- El número de iteraciones alcanza o supera el número máximo de iteraciones predefinido.

En la práctica, las iteraciones continúan hasta que se satisface alguna combinación de criterios de parada.

Ambigüedades

Como se vio en el apartado 2.2.5, generalmente la estimación en NMF se realiza mediante la minimización de una o varias funciones objetivos. Sin embargo, en general, estas minimizaciones no garantizan una solución única. Incluso la función cuadrática con respecto a ambos conjuntos de argumentos $\{\mathbf{A}\}$ y $\{\mathbf{X}\}$ puede tener varios mínimos locales, lo que hace que los algoritmos NMF puedan sufrir indeterminaciones rotacionales (ambigüedades).

Debido a estas indeterminaciones o ambigüedades, es posible que la solución a NMF no sea única, por ejemplo, considerando la siguiente ecuación cuadrática:

$$D_F(\mathbf{Y} \parallel \mathbf{AX}) = \parallel \mathbf{Y} - \mathbf{AX} \parallel_F^2 = \parallel \mathbf{Y} - \mathbf{AR}^{-1} - \mathbf{RX} \parallel_F^2 = \parallel \mathbf{Y} - \tilde{\mathbf{A}}\tilde{\mathbf{X}} \parallel_F^2, \tag{2.55}$$

donde la matriz rotacional $\mathbf{R}$, ha de escogerse de manera que las matrices transformadas $\tilde{\mathbf{A}} \neq \mathbf{A}$ y $\tilde{\mathbf{X}} \neq \mathbf{X}$ sean no negativas. Es importante resaltar, que la inversa de una matriz no negativa es no negativa, si y solo si, esta es una matriz de permutación generalizada. Una matriz de permutación generalizada es aquella que tiene un solo elemento positivo distinto de cero en cada fila y en cada columna. Si asumimos que $\mathbf{R} \geq 0$ y $\mathbf{R}^{-1} \geq 0$, las cuales son condiciones suficientes para la no negatividad de las matrices $\mathbf{AR}^{-1}$ y $\mathbf{RX}$, entonces $\mathbf{R}$ tiene que ser una matriz de permutación generalizada, es decir, $\mathbf{R}$ se puede expresar como el producto de una matriz diagonal no singular definida positiva y una matriz de permutación. Si las matrices originales $\mathbf{X}$ y $\mathbf{A}$ están suficientemente dispersas solo una matriz de permutación $\mathbf{P} = \mathbf{R}$ puede satisfacer la restricción de no negatividad de cualquier matriz de transformación y obtener una NMF única.

Para ilustrar la indeterminación rotacional descrita, se expone el siguiente ejemplo, dadas las siguientes matrices de mezcla y fuente:

$$\mathbf{A} = \begin{vmatrix} 3 & 2 \\ 7 & 2 \end{vmatrix}, \qquad \mathbf{X} = \begin{vmatrix} x_1(t) \\ x_2(t) \end{vmatrix}$$

su producto da como resultado:

$$\mathbf{Y} = \begin{vmatrix} \mathbf{y}_1(t) \\ \mathbf{y}_2(t) \end{vmatrix} = \mathbf{AX} \begin{vmatrix} 3\mathbf{x}_1(t) + 2\mathbf{x}_2(t) \\ 7\mathbf{x}_1(t) + 2\mathbf{x}_2(t) \end{vmatrix}$$

No obstante, es evidente que exiten otras descomposiciones no negativas que dan el mismo resultado, por ejemplo:

$$\mathbf{Y} = \begin{vmatrix} \mathbf{y}_1(t) \\ \mathbf{y}_2(t) \end{vmatrix} = \tilde{\mathbf{A}}\tilde{\mathbf{X}} = \begin{vmatrix} 0 & 1 \\ 4 & 1 \end{vmatrix} \begin{vmatrix} \mathbf{x}_1(t) \\ 3\mathbf{x}_1(t) + \mathbf{x}_2(t) \end{vmatrix},$$

donde:

$$\tilde{\mathbf{A}} = \begin{vmatrix} 0 & 1 \\ 4 & 1 \end{vmatrix}, \qquad \tilde{\mathbf{X}} = \begin{vmatrix} \mathbf{x}_1(t) \\ 3\mathbf{x}_1(t) + \mathbf{x}_2(t) \end{vmatrix}$$

son nuevos componentes no negativos que no provienen de las indeterminaciones de permutación o escalado.

Sin embargo, incorporando alguna medida de dispersión o suavidad a la función objetivo, es suficiente para resolver el problema NMF de forma única.

Cuando no hay información a priori disponible, debemos normalizar las columnas de $\mathbf{A}$ y/o las filas de $\mathbf{X}$ para ayudar a mitigar los efectos de las indeterminaciones de rotación. Dicha normalización se suele hacer escalando las columnas a_j de $\mathbf{A} = [a_1, \ldots, a_J]$ de la siguiente manera:

$$\mathbf{A} \leftarrow \mathbf{AD}_A, \quad \text{donde} \quad D_A = \operatorname{diag}\left(\parallel a_1 \parallel_p^{-1}, \parallel a_2 \parallel_p^{-1}, \ldots, \parallel a_J \parallel_p^{-1} \right), \quad p \in [0, \infty)$$

Varios experimentos demuestran que, los mejores resultados se obtienen para $p = 1$. Por otro lado, para evitar las indeterminaciones rotacionales, las filas de $\mathbf{X}$ deben ser dispersas o tener nivel de referencia cero, como

ocurre, por ejemplo, en NMF con offset (ver punto 2.2.3). En definitiva, a fin de obtener una única solución de NMF, es necesario llevar a cabo alguna de las siguientes técnicas:

- Normalizar o filtrar la matriz de datos de entrada $\mathbf{Y}$.
- Normalizar las columnas de $\mathbf{A}$ y/o las filas de $\mathbf{X}$.
- Imponer condiciones de dispersión y/o suavidad a las matrices factorizadas.

NMF a gran escala (Large-Scale NMF)

En algunos casos, especialmente en los de reducción de la dimensión, la matriz de datos $\mathbf{Y} \in \mathbb{R}^{I \times T}$ puede llegar a tener una gran dimensión (del orden de millones de entradas), pero puede ser factorizada aproximadamente usando un número considerablemente menor de componentes no negativos (J), es decir, $J << I$ y $J << T$. Entonces el problema $\mathbf{Y} \approx \mathbf{AX}$ se vuelve muy redundante y por tanto, no tenemos que usar la información de todas las entradas de $\mathbf{Y}$ para estimar de forma precisa las matrices $\mathbf{A} \in \mathbb{R}^{I \times J}$ y $\mathbf{X} \in \mathbb{R}^{J \times T}$. En otras palabras, para resolver el problema de NMF a gran escala, no necesitamos conocer la matriz de datos $\mathbf{Y}$ completa, si no que nos basta con conocer una pequeña parte aleatoria, ni tampoco tenemos que realizar cálculos en cada iteración de las matrices estimadas $\mathbf{Y}^T\mathbf{A}$ o $\mathbf{YX}^T$. Este enfoque puede superar considerablemente a los métodos estándar de NMF, especialmente para sistemas extremadamente sobredeterminados.

De esta forma, para representar NMF básica $\mathbf{Y} = \mathbf{AX} + \mathbf{E}$, se consideran dos descomposiciones no negativas utilizando matrices de rango mucho más pequeño, dadas por:

$$\begin{aligned} \mathbf{Y}_r &= \mathbf{A}_r\mathbf{X} + \mathbf{E}_r, \quad \text{para} \quad \mathbf{A}_r \quad \text{fijo y conocido} \\ \mathbf{Y}_c &= \mathbf{AX}_c + \mathbf{E}_c, \quad \text{para} \quad \mathbf{X}_c \quad \text{fijo y conocido} \end{aligned} \tag{2.56}$$

donde $\mathbf{Y}_r \in \mathbb{R}_+^{R \times T}$ y $\mathbf{Y}_c \in \mathbb{R}_+^{I \times C}$ son las matrices construidas con las filas y columnas elegidas de la matriz $\mathbf{Y}$ respectivamente. De manera análoga, pueden construirse las matrices reducidas: $\mathbf{A}_r \in \mathbb{R}_+^{R \times J}$ y $\mathbf{X}_c \in \mathbb{R}_+^{J \times C}$. Normalmente, es suficiente con tomar los índices de forma que: $J < R \leq 4J$ y $J < C \leq 4J$.

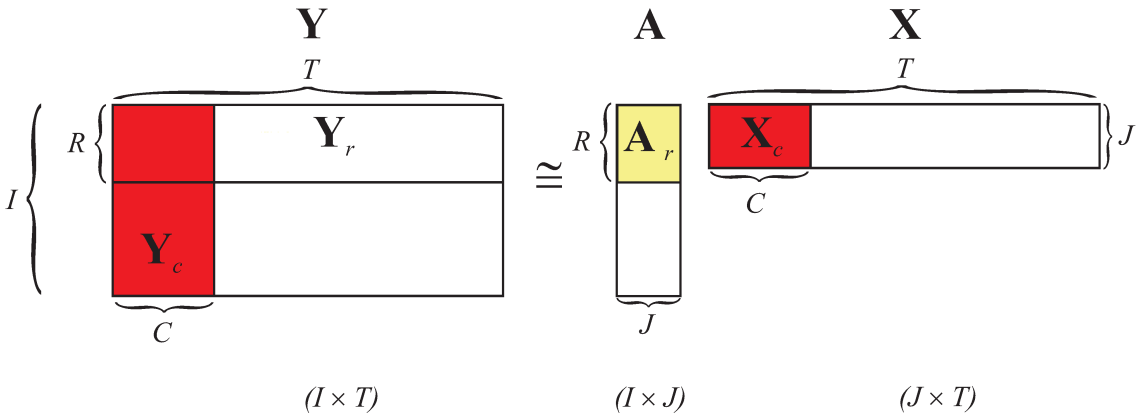

Figura 2.14 Ilustración conceptual del procesado de datos en bloques para NMF a gran escala. En vez de procesar toda la matriz $\mathbf{Y} \in \mathbb{R}^{I \times T}$, podemos procesar bloques de matrices mucho más pequeños $\mathbf{Y}_c \in \mathbb{R}^{I \times C}$ y $\mathbf{Y}_r \in \mathbb{R}^{R \times T}$ y las matrices de factorización correspondientes $\mathbf{X}_c \in \mathbb{R}^{J \times C}$ y $\mathbf{A}_r \in \mathbb{R}^{R \times J}$ con $C << T$ y $R << I$. Para simplificar la ilustración gráfica, hemos asumido que se seleccionan las primeras R filas y C columnas de las matrices $\mathbf{Y}$,$\mathbf{A}$ y $\mathbf{X}$ [12].

Existen varias estrategias para elegir los elementos de la matriz de entrada de datos, el escenario más simple es seleccionar las R primera filas y las C primeras columnas de $\mathbf{Y}$ como se muestra en la Figura 2.14. Otras estrategias consisten en elegirlos mediante una función de distribución uniforme, tomar de manera aleatoria columnas y filas con probabilidades proporcionales a su importancia, o seleccionar aquellas que tienen una norma mayor.

2.2.7 NMF en la separación de fuentes de audio

Desde que se hiciera popular, NMF ha sido uno de los métodos más usados para la resolución de problemas de separación de fuentes de audio.

Las señales de audio se representan en el dominio del tiempo-frecuencia mediante una Transformada Localizada de Fourier (Short Term Fourier Transform, STFT) de valores complejos. El problema aparece cuando se quiere estimar S_1 y S_2 de una mezcla dada de dos fuentes expresada como [19]:

$$X = S_1 + S_2, \qquad X, S_1, S_2, \in \mathbb{C}^{F \times N}, \tag{2.57}$$

En el enfoque básico de resolución de este problema mediante NMF se haría [19]:

1. Calculando una descomposición NMF tal que $V = \mid X \mid^2 \approx WH = W_1H_1 + W_2H_2$.
2. Calculando las estimaciones de las fuentes mediante un filtro de Wiener:
   $\hat{S_1} = \frac{W_1H_1}{W_1H_1+W_2H_2} \bigodot X, \qquad \hat{S_2} = \frac{W_2H_2}{W_1H_1+W_2H_2} \bigodot X.$

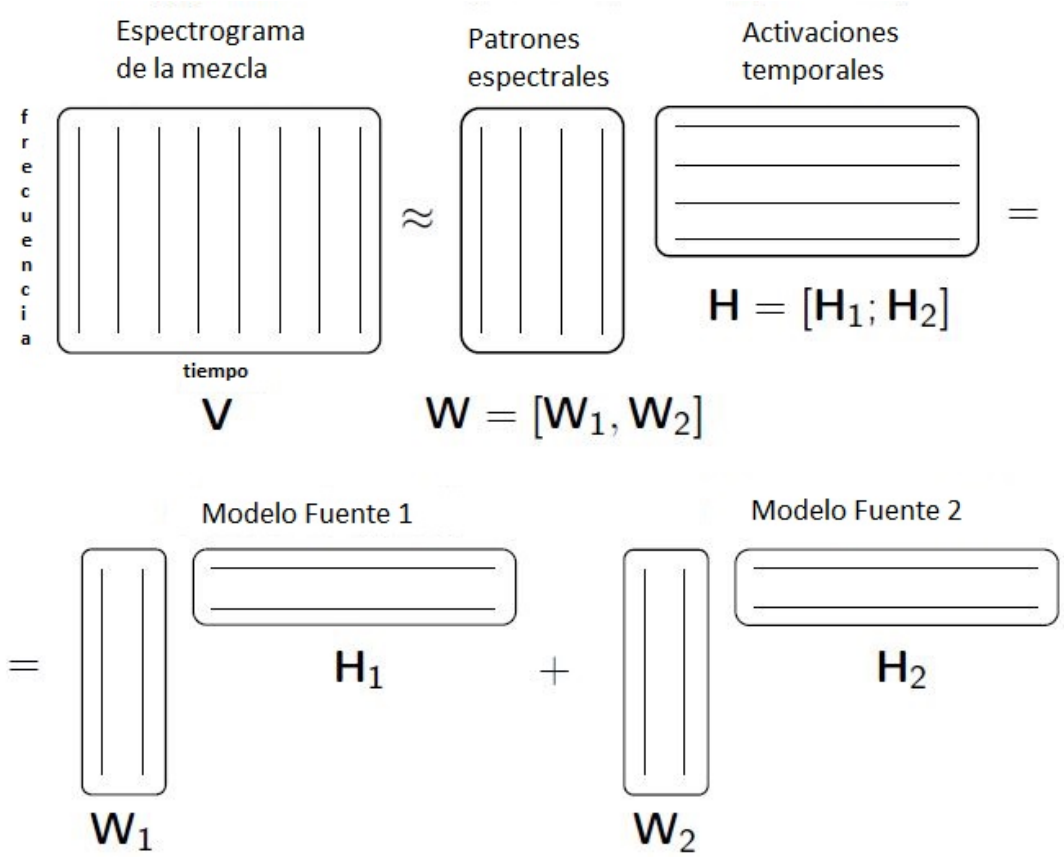

Figura 2.15 Esquema básico de la separación de fuentes de audio mediante NMF [19].

La principal dificultad a la que nos enfrentamos en este problema es la de calcular la descomposición $V \approx WH = W_1H_1 + W_2H_2$ tal que (W_1, H_1) y (W_2, H_2) representen bien las fuentes S_1 y S_2 respectivamente.

Una de las aproximaciones más populares se basa en calcular W_1 y W_2 de algunas muestras entrenadas con un algoritmo iterativo. Se establece $W = [W_1, W_2]$ y se van actualizando en cada iteración de manera eficiente durante la descomposición de la mezcla.

3 Modelo de Destino Recurrente (Common Fate Model, CFM)

En [53], *Stöter et al.* propusieron un método de BSS para superar la dificultad que entraña modelar señales no estacionarias. El método, puede ser aplicado a mezclas de diferentes instrumentos musicales con modulaciones en frecuencia y/o amplitud, en este caso, estas modulaciones son provocadas por un vibrato. El modelo se basa en una representación de señal que divide el espectrograma complejo en una rejilla de parches de tamaño arbitrario. Estos parches complejos son procesados con una transformada de Fourier bidimensional discreta, formando una representación tensorial que revela las texturas de la modulación temporal y espectral. Esta representación se puede ver como una alternativa a las transformadas de modulación calculadas en espectrogramas de magnitud. Un modelo de factorización adaptado, permite descomponer diferentes fuentes de armónicos variables en el tiempo, basándose en sus perfiles de modulación recurrente: de ahí el nombre *Modelo de Destino Recurrente*.

3.1 Introducción

Como se comenta en la sección 2.1, la separación de fuentes de audio continúa siendo un campo de investigación muy activo. En [53], se estudia un caso muy concreto donde dos de las suposiciones más importantes de NMF, para separación de fuentes de audio, dejan de ser válidas. La primera suposición es que los armónicos espectrales solo se solapan parcialmente, cosa que no se puede asumir en este caso, donde ambos instrumentos tocan la misma nota, o lo que es lo mismo, comparten la misma frecuencia fundamental. La segunda suposición, es que todas las matrices temporales y espectrales de NMF corresponden a notas musicales, formando así un diccionario de entradas con sentido musical. Este hecho, no se cumple cuando un instrumento tiene fluctuaciones variables en el tiempo, como en este caso, que ambos instrumentos ejecutan un vibrato, produciéndose así, fluctuaciones tanto temporales como de amplitud. Así que con la imposibilidad de tomar estas suposiciones, éste se convierte en un escenario especialmente desafiante [54].

En este trabajo, seguimos el modelo tensorial presentado en [53], el cual explota las similitudes en frecuencia. Este modelo, también nos permite hacer uso de las dependencias entre las modulaciones de los intervalos vecinos. Se aprecian ciertas coincidencias con el modelo HR-NMF, del cual se habla en la Sección 2.2.4 y que tiene en cuenta las dependencias en el plano tiempo-frecuencia. La idea principal es dividir el espectrograma complejo en parches de modulación para agrupar la modulación recurrente. A esto se le ha llamado *Modelo de Destino Recurrente* (Common Fate Model, CFM), tomándolo prestado de la teoría de Gestalt, la cual describe cómo la percepción humana fusiona objetos que se mueven juntos en el tiempo.

3.2 Modelado del Destino Recurrente

3.2.1 Transformada de Destino Recurrente (Common Fate Transform, CFT)

Si llamamos $\tilde{x}$ a una señal de audio mono canal. Su Transformada Localizada de Fourier (STFT) se calcula dividiéndola en cuadros (frames) solapados y después haciendo la Transformada Discreta de Fourier (DFT) a cada cuadro. La información obtenida es recopilada en una matriz de $N_\omega \times N_\tau$ a la que llamamos $\mathbf{X}$, donde N_ω es el número de bandas de frecuencias y N_τ el número total de cuadros. Siguiendo el trabajo [53], se considerará las propiedades de otro objeto, construido desde $\mathbf{X}$, al que han llamado la Transformada de Destino Recurrente (CFT). Éste es construido tal como se ilustra en la Figura 3.3 y de forma más gráfica en la Figura 3.2.

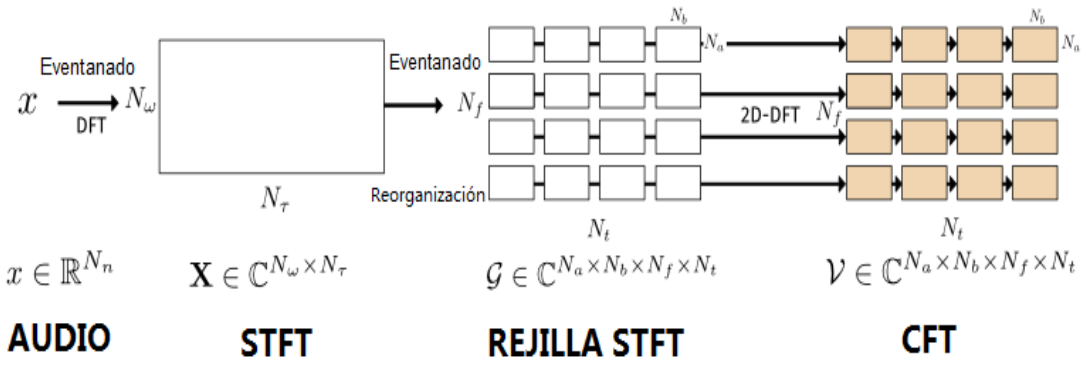

Figura 3.1 Transformada de Destino Recurrente, CFT [53].

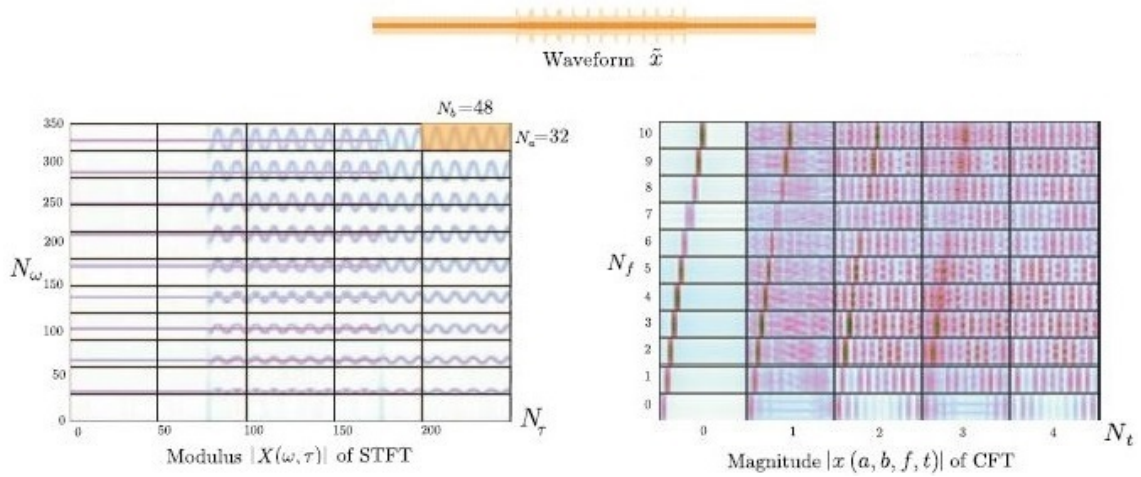

Figura 3.2 Transformada de Destino Recurrente,CFT. Para una mejor representación, la división de la STFT en parches se ha representado sin solape, pero en la práctica se usan parches solapados [53].

Dividimos la STFT de $\mathbf{X}$ en parches rectangulares solapados de tamaño $N_a \times N_b$, regularmente espaciados tanto en tiempo como en frecuencia. Después, se calcula la 2D-DFT de cada parche. Esto produce un tensor de $N_a \times N_b \times N_f \times N_t$ que se escribe como $\mathbf{x}$, donde N_f y N_t son las posiciones verticales y horizontales de los parches respectivamente.

La CFT es básicamente otra 2D-DFT tomada de la STFT estándar de $\mathbf{X}$, se calcula usando la STFT compleja de $\mathbf{X}$, y no una representación de la magnitud como $|\mathbf{X}|$. Una de las propiedades esenciales para que la CFT se pueda considerar válida en el campo de la separación, es que es invertible: la forma de onda original $\tilde{x}$ puede recuperarse de forma exacta.

3.2.2 Modelo probabilístico de la CFT

El modelo probabilístico de la CFT está extensamente explicado en [53], debido a que en este trabajo se ha seguido dicho modelo, solo vamos a resumir las 4 suposiciones principales:

1 *Todos los parches son independientes*. En [53], se asume la independencia de los parches solapados. Debido al solape entre los parches, esta suposición es una aproximación.

2 *Cada parche es estacionario*: su distribución no depende de las translaciones en el plano tiempo-frecuencia. Aquí es donde [53] no asume independencia, sino que espera dependencias entre las entradas vecinas de la STFT. La diferencia con el modelo HR-NMF (del que se habla en la Sección 2.2.4) es que tenemos innovaciones independientes e idénticamente distribuidas para cada parche dado, mientras que el modelo HR-NMF tiene más variabilidad.

3 *La distribución conjunta de todas las entradas de cada parche es* $\alpha-estable$ [50].

4 *Cada parche es armonizable*.

Bajo estas cuatro suposiciones, todas las entradas de la CFT $\mathbf{x}$ son independientes (suposiciones 1 y 2), y cada una distribuida en base a una distribución isotrópica compleja $\alpha-estable$, denotada $S_\alpha S_c$ (como supuestos 3 y 4):

$$\mathbf{x}(a,b,f,t) \sim S_\alpha S_c(\mathbf{P}^\alpha(a,b,f,t)), \tag{3.1}$$

donde $\mathbf{P}^\alpha$ es un tensor no negativo de $N_a \times N_b \times N_f \times N_t$ al que llamamos *densidad de modulación*. En el caso general, esto puede entenderse como la energía encontrada en (a,b) para el parche (f,t).

3.2.3 Separación de señales

Se asume en [53], que la forma de onda observada es realmente la suma de J fuentes subyacentes $\{\tilde{s}_j\}_{j=1,\dots,J}$, en este trabajo se ha estudiado solo el caso de $J = 2$ aunque es posible aumentar este número. Debido a la linealidad de la CFT, esto puede expresarse en el dominio de la CFT como:

$$\forall(a,b,f,t), \mathbf{x}(a,b,f,t) = \sum_j \mathbf{s}_j(a,b,f,t).$$

Si adoptamos el modelo $\alpha-estable$ presentado anteriormente para cada fuente y usamos la propiedad de estabilidad, tendremos:

$$\mathbf{x}(a,b,f,t) \sim S_\alpha S_c \left(\sum_j \mathbf{P}_j^\alpha(a,b,f,t) \right),$$

donde P_j^α es la densidad de modulación de la fuente j. Si estos objetos son conocidos, se puede demostrar que cada fuente puede ser estimada en un sentido máximo a posteriori de la mezcla como:

$$\mathbb{E}\left[\mathbf{s}_j(a,b,f,t) \mid \{\mathbf{P}_j^\alpha\}_j, x\right] = \frac{\mathbf{P}_j^\alpha(a,b,f,t)}{\sum_{j'} \mathbf{P}_{j'}^\alpha(a,b,f,t)} \mathbf{x}(a,b,f,t) \tag{3.2}$$

que es llamado filtro $\alpha-Wiener$ fraccionario [38]. La forma de onda resultante se obtiene invirtiendo la CFT. Como puede observarse, ahora es necesario estimar las densidades de modulación $\{\mathbf{P}_j^\alpha\}_j$ basándonos en la observación de la mezcla CFT $\mathbf{x}$.

3.2.4 Modelo de factorización y estimación de los parámetros

En primer lugar, imponemos el modelo de factorización de [53] sobre las fuentes, para reducir el número de parámetros a estimar. En [53] se fijan:

$$\mathbf{P}_j^\alpha(a,b,f,t) = \mathbf{A}_j(a,b,f) H_j(t), \tag{3.3}$$

donde $\mathbf{A}_j$ es un tensor de dimensiones $N_a \times N_b \times N_f$ y $\mathbf{H}_j$ es un vector de dimensión $N_t \times 1$, ambos no negativos. A esto se le llama el *Modelo de Destino Recurrente* [53]. $\mathbf{A}_j$ contiene las densidades de modulación, que son diferentes para cada banda de frecuencia, y eso capta el perfil de modulación a largo plazo de la fuente j alrededor de esa frecuencia. $\mathbf{H}_j$ es un vector de activación que indica la potencia de la fuente j en los parches situados en la posición temporal t. El aprendizaje de esos parámetro podría haberse hecho usando el método

convencional de Factorización No Negativa de Matrices (NMF, ver Sección 2.2) excepto que se aplica a la CFT en vez de a la STFT, así que la factorización que se va a utilizar es (3.3).

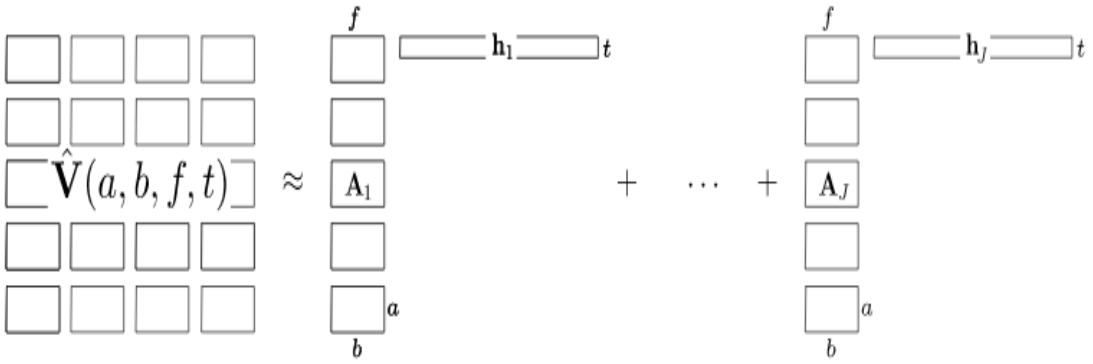

Figura 3.3 Modelo de Destino Recurrente, CFM [53].

Básicamente, lo que se busca es estimar los parámetros $\{\mathbf{A}_j, \mathbf{H}_j\}$ para que el módulo de la CFT, elevado a la potencia α , esté lo más cerca posible de $\sum_j \mathbf{P}_j^{\alpha}$, con alguna función de coste particular como criterio de ajuste de datos, llamada $\beta-$*divergencia* que incluye casos especiales como la Euclídea, Kullback-Leibler e Itakura-Saito [21]. Como es común en los modelos no negativos, cada parámetro es actualizado por turno, mientras que otros se dejan fijos. Se proporcionan las actualizaciones multiplicativas en el Algoritmo 1. Después de varias iteraciones, los parámetros pueden usarse en (3.2) para separar las fuentes.

Algoritmo 1: Ajuste de los parámetros NMF de la CFM no negativa(3.3) [53]

Con $\upsilon^{\alpha} = |\mathbf{x}|^{\alpha}$ y usando siempre los parámetros actualizados para calcular $\hat{\mathbf{P}}^{\alpha}(a,b,f,t) = \sum_{j=1}^{J} \mathbf{A}_j(a,b,f)\mathbf{H}_j(t)$, iterar:

$$\mathbf{A}_j(a,b,f) \leftarrow \mathbf{A}_j(a,b,f)\frac{\sum_t \upsilon^{\alpha}(a,b,f,t)\hat{\mathbf{P}}^{\alpha}(a,b,f,t)^{\cdot(\beta-2)}\mathbf{H}_j(t)}{\sum_t \hat{\mathbf{P}}^{\alpha}(a,b,f,t)^{\cdot(\beta-1)}\mathbf{H}_j(t)}$$

$$\mathbf{H}_j(t) \leftarrow \mathbf{H}_j(t)\frac{\sum_{a,b,f} \upsilon^{\alpha}(a,b,f,t)\hat{\mathbf{P}}^{\alpha}(a,b,f,t)^{\cdot(\beta-2)}\mathbf{A}_j(a,b,f)}{\sum_{a,b,f} \hat{\mathbf{P}}^{\alpha}(a,b,f,t)^{\cdot(\beta-1)}\mathbf{A}_j(a,b,f)}$$

3.3 Estudio de las Alfa-Beta divergencias

Una vez estudiado e implementado el modelo, en la parte práctica de este trabajo, se hizo un estudio de cómo afectaban los valores de los parámetros α y β a los resultados de la separación, es decir, cómo afectan a la Relación Señal a Ruido (SDR), a la Relación Señal a Artefacto (SAR) y a la Relación Señal a Interferencia (SIR). No vamos a entrar en los detalles prácticos en esta sección ya que se exponen en el siguiente capítulo, pero vamos a introducir el concepto de Alfa-Beta divergencia en la separación de fuentes de audio. Los parámetros α y β están relacionados con el concepto de Alfa-Beta divergencia que fue propuesto en [12].

3.3.1 Definición de Alfa-Beta divergencia

Sean las matrices positivas $\mathbf{P}$ y $\mathbf{Q}$ de dimensiones $I \times T$ y, p_{it} y q_{it} las entradas de dichas matrices, se define la divergencia alfa-beta o divergencia AB, como la medida de similitud entre ambas matrices, de la siguiente manera:

$$D_{AB}^{(\alpha,\beta)}(\mathbf{P} \parallel \mathbf{Q}) = -\frac{1}{\alpha\beta}\sum_{it}\left(p_{it}^{\alpha}q_{it}^{\beta} - \frac{\alpha}{\alpha+\beta}p_{it}^{\alpha+\beta} - \frac{\beta}{\alpha+\beta}q_{it}^{\alpha+\beta}\right) \quad \text{para } \alpha,\beta,\alpha+\beta \neq 0 \tag{3.4}$$

O de forma equivalente:

$$D_{AB}^{\alpha,\lambda-\alpha}(\mathbf{P}\parallel\mathbf{Q})=\frac{1}{(\alpha-\lambda)\,\alpha}\sum_{it}\left(p_{it}^{\lambda}q_{it}^{\lambda-\alpha}-\frac{\alpha}{\lambda}p_{it}^{\lambda}-\frac{\lambda-\alpha}{\lambda}q_{it}^{\lambda}\right)\quad\text{para }\alpha\neq0,\alpha\neq\lambda,\lambda=\alpha+\beta\neq0. \tag{3.5}$$

Para evitar indeterminaciones o singularidades para algunos valores de los parámetros, la divergencia AB puede extenderse por continuidad (aplicando la fórmula de l'Hôpital) para cubrir todos los valores de $\alpha,\beta\in\mathbb{R}$, así la divergencia AB puede expresarse de una forma más explícita:

$$D_{AB}^{(\alpha,\beta)}(\mathbf{P}\parallel\mathbf{Q})=\sum_{it}d_{AB}^{(\alpha,\beta)}(p_{it},q_{it}), \tag{3.6}$$

donde

$$d_{AB}^{(\alpha,\beta)}(p_{it}q_{it})=\begin{cases}-\frac{1}{\alpha\beta}\left(p_{it}^{\alpha}q_{it}^{\beta}-\frac{\alpha}{(\alpha+\beta)}p_{it}^{\alpha+\beta}-\frac{\beta}{(\alpha+\beta)}q_{it}^{\alpha+\beta}\right) & \text{para}\quad\alpha,\beta,\alpha+\beta\neq0\\ \frac{1}{\alpha^2}\left(p_{it}^{\alpha}\ln\frac{p_{it}^{\alpha}}{q_{it}^{\alpha}}-p_{it}^{\alpha}+q_{it}^{\alpha}\right) & \text{para}\quad\alpha\neq0,\beta=0\\ \frac{1}{\alpha^2}\left(\ln\frac{q_{it}^{\alpha}}{p_{it}^{\alpha}}+\left(\frac{q_{it}^{\alpha}}{p_{it}^{\alpha}}\right)^{-1}-1\right) & \text{para}\quad\alpha=-\beta\neq0\\ \frac{1}{\beta^2}\left(q_{it}^{\beta}\ln\frac{q_{it}^{\beta}}{p_{it}^{\beta}}-q_{it}^{\beta}+p_{it}^{\beta}\right) & \text{para}\quad\alpha=0,\beta\neq0\\ \frac{1}{2}\left(\ln p_{it}-\ln q_i t\right)^2 & \text{para}\quad\alpha,\beta=0\end{cases} \tag{3.7}$$

Sustituyendo los parámetros α y β con los valores adecuados, pueden obtenerse otras distancias conocidas, como la divergencia de Kullback-Leibler (para $\alpha=1$ y $\beta=0$), o la divergencia de Itakura-Saito (para $\alpha=1$ y $\beta=-1$), entre otras

$$\begin{aligned}D_{AB}^{(1,0)}(\mathbf{P}\parallel\mathbf{Q})&=D_{KL}(\mathbf{P}\parallel\mathbf{Q})=\left(p_{it}\ln\frac{p_{it}}{q_{it}}-p_{it}+q_{it}\right)\\ D_{AB}^{(1,-1)}(\mathbf{P}\parallel\mathbf{Q})&=D_{IS}(\mathbf{P}\parallel\mathbf{Q})=\left(\ln\frac{p_{it}}{q_{it}}+\frac{q_{it}}{p_{it}}-1\right)\end{aligned} \tag{3.8}$$

Por otro lado, particularizando para ciertos valores, se obtienen otras divergencias como la Alfa (para $\alpha+\beta=1$), o la divergencia Beta (para $\alpha=0$). Por lo tanto, podemos decir que la divergencia AB es una medida de similitud general, a partir de la cual es posible obtener muchas de las divergencias más utilizadas.

3.3.2 Propiedades

Considerando el operador $\mathbf{P}^{[r]}$ como la transformación que eleva todos los elementos de la matriz $\mathbf{P}$ al valor de r, es decir, p_{it}^{r}, se definen las siguientes propiedades de la divergencia AB:

- *Dualidad*: esta propiedad implica que una permutación en los parámetros, provoca una permutación en las matrices.
$$D_{AB}^{(\alpha,\beta)}(\mathbf{P}\parallel\mathbf{Q})=D_{AB}^{(\alpha,\beta)}(\mathbf{Q}\parallel\mathbf{P}) \tag{3.9}$$

- *Inversión*: un cambio de signo en los parámetros, se traduce en la inversión de los elementos de las matrices:
$$D_{AB}^{(-\alpha,-\beta)}(\mathbf{P}\parallel\mathbf{Q})=D_{AB}^{(\alpha,\beta)}\left(\mathbf{P}^{[-1]}\parallel\mathbf{Q}^{[-1]}\right) \tag{3.10}$$

- *Escalado de parámetros*:las propiedades anteriores pueden ser consideradas como casos particulares de la propiedad de escalado de los parámetros α y β por un factor común $\omega\in\mathbb{R}\backslash\{0\}$. La divergencia cuyos parámetros han sido re-escalados es proporcional a la divergencia original con ambos argumentos elevados al factor común, es decir:
$$D_{AB}^{(\omega\alpha,\omega\beta)}(\mathbf{P}\parallel\mathbf{Q})=\frac{1}{\omega^2}D_{AB}^{(\alpha,\beta)}\left(\mathbf{P}^{\cdot[\omega]}\parallel\mathbf{Q}^{\cdot[\omega]}\right). \tag{3.11}$$
Esta propiedad puede verse como un "zoom-in" a los argumentos de $\mathbf{P}$ y $\mathbf{Q}$ cuando $\omega<1$. Dicho "zoom" da más relevancia a los valores pequeños frente a los mayores. Al contrario, cuando $\omega>1$ se

produce un efecto "zoom-out" donde los valores pequeños pierden relevancia en detrimento de los valores grandes (ver Figura 3.4).

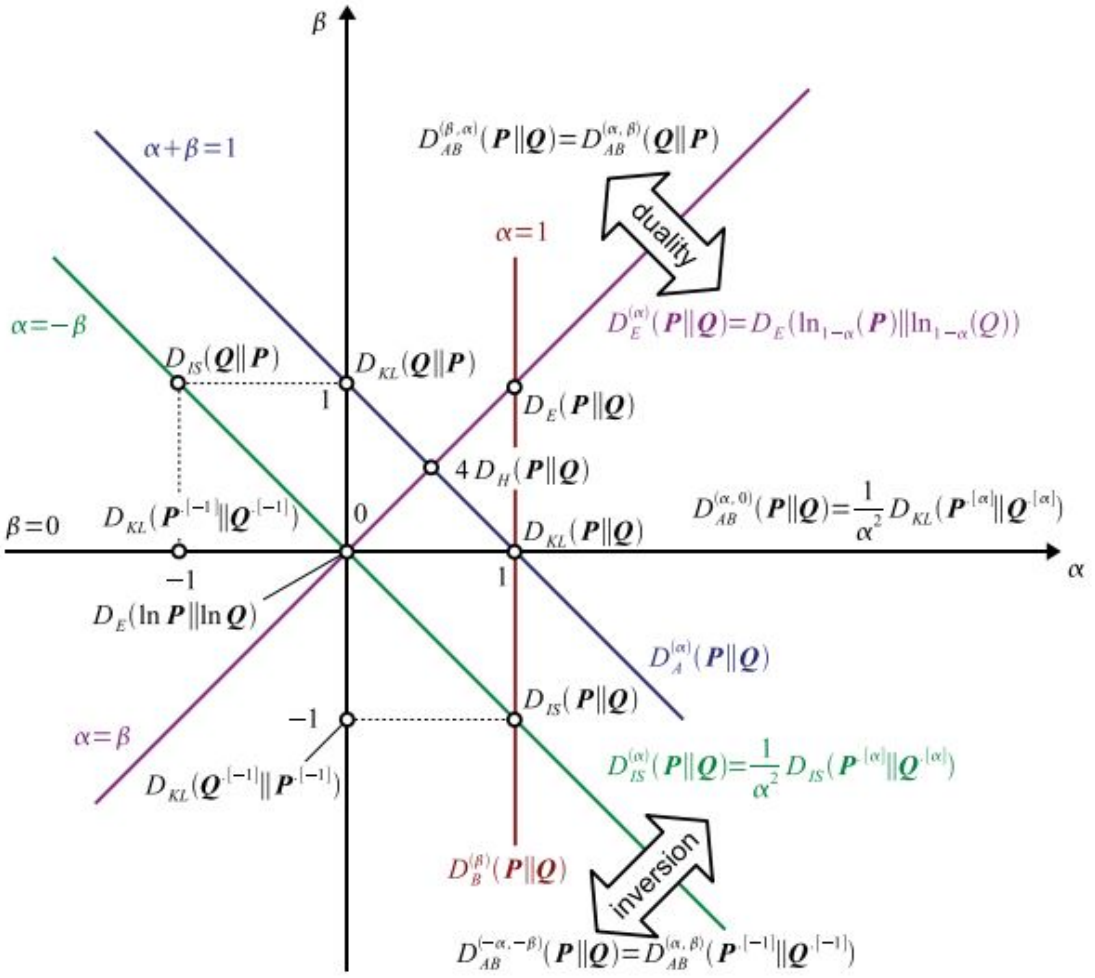

Figura 3.4 Ilustración gráfica de las propiedades de inversión y dualidad en la divergencia-AB. En el plano $\alpha-\beta$ están indicados como casos importantes divergencias particulares con puntos y líneas, especialmente la divergencia Kullback-Leibler D_{KL}, Distancia de Hellinger D_H, distancia Euclídea D_E, distancia de Itakura-Saito D_{IS}, Alfa-divergencia D_A^{α}, y Beta-divergencia $D_B^{(\beta)}$ [12].

Todas estas propiedades permiten reformular la definición de la divergencia AB y expresarla de distintas formas, por ejemplo en términos de otras divergencias combinadas con zooms de los parámetros.

3.3.3 Justifcación del estudio

El hecho expuesto anteriormente y en [12], que implica que la divergencia AB incluye varias de las divergencias más populares en función del valor de los parámetros α y β, hace que esta divergencia sea muy interesante para examinar la eficiencia de nuestro algoritmo ya que da una gran versatilidad a la solución. También resulta muy interesante para resolver problemas de NMF ya que la divergencia AB cumple la restricción de no negatividad, al cumplirla las divergencias que abarca.

Por otro lado, la divergencia AB es notablemente más robusta que otras, frente a los errores y al ruido, gracias al uso de los hiperparámetros α y β. El modelo factorizado puede verse como una función vectorial de una serie de parámetros θ, donde cada uno de sus elementos $q_{it}(\theta) > 0$ es no negativo para un rango de parámetros determinado. Entonces, el estimador $\hat{\theta}$ entre dos medidas discretas positivas **P** y **Q**, para la

divergencia AB, es una solución de la ecuación:

$$\frac{\partial D_{AB}^{(\alpha,\beta)}(\mathbf{P}\parallel\mathbf{Q})}{\partial\theta} = -\sum_{it}\frac{\partial q_{it}}{\partial\theta}q_{it}^{\alpha+\beta-1}\ln_{1-\alpha}(p_{it}/q_{it}) = 0, \tag{3.12}$$

donde la función $\ln_{1-\alpha}$ es el logaritmo deformado, definido como:

$$ln_{1-\alpha}(z) = \begin{cases} \frac{z^{\alpha}-1}{\alpha}, & \text{si} \quad \alpha \neq 0 \\ \ln(z), & \text{si} \quad \alpha = 0 \end{cases}$$

Atendiendo a la ecuación 3.12, puede verse la influencia que tiene cada parámetro sobre la estimación. En el caso de α, se controlan los valores individuales de los cocientes p_{it}/q_{it}, lo que puede ser interpretado como un "zoom", donde se dará más importancia a valores altos, en el caso de $\alpha > 1$, o por el contrario a valores pequeños, si $\alpha < 1$. De igual forma, el parámetro β puede controlar la influencia del cociente p_{it}/q_{it}, normalmente se buscan valores que permitan un buen compromiso entre la robustez, para valores $\beta > 1$, y la eficiencia, para $\beta = 0$. En definitiva, el parámetro α, controla la influencia de los cocientes en el estimador, mientras que β, controla la ponderación de dichos cocientes dependiendo de los valores que mejor se ajusten al modelo.

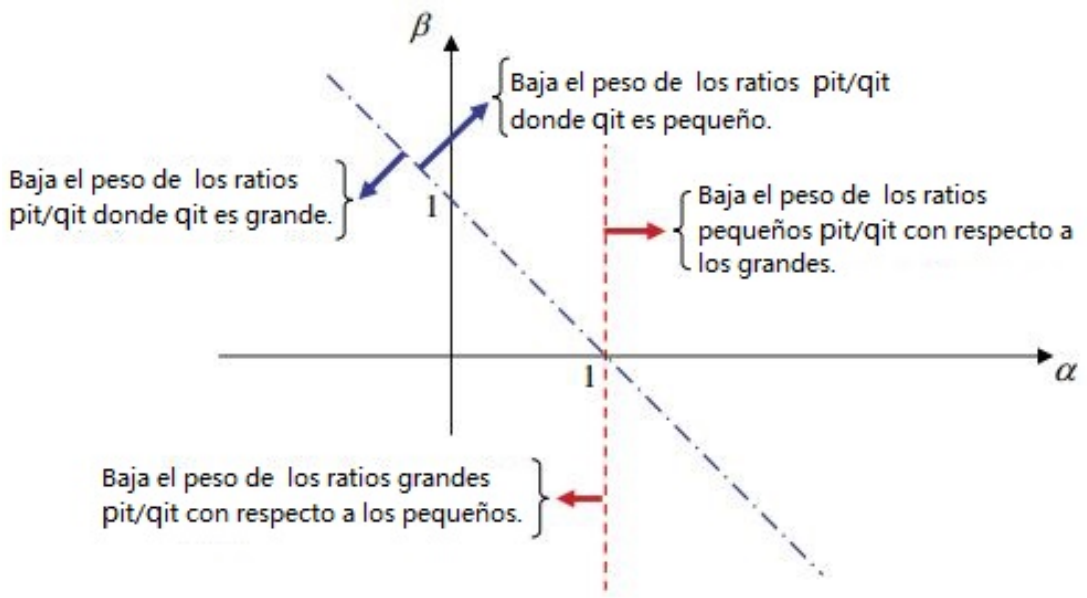

Figura 3.5 Ilustración gráfica de cómo los parámetros establecidos α y β pueden controlar la influencia de los ratios individuales p_{it}/q_{it}. La línea de puntos y rayas ($\alpha+\beta=1$) muestra la región donde el factor de ponderación multiplicativo $q_{it}^{\alpha+\beta-1}$ en la ecuación de estimación es constante y unitario. La línea de rayas ($\alpha = 1$) muestra la región donde el orden del logaritmo deformado de p_{it}/q_{it} es constante e igual al de la divergencia Kullback-Leibler estándar [12].

Resumiendo, la elección de la divergencia AB para este estudio se justifica gracias a su no negatividad, lo que permite que pueda usarse para resolver el problema de NMF, su versatilidad y su eficiencia, así como su robustez frente a ruidos y errores.

4 Simulaciones

En este capítulo, se presenta todo el trabajo práctico realizado en Matlab® para implementar el modelo expuesto en el Capítulo 3 y los resultados de las diversas simulaciones realizadas, con el fin de exponer el trabajo elaborado.

4.1 Datos de entrada

Para poder comparar nuestras simulaciones con las de [53], se ha usado el mismo dataset, que consta de 10 mezclas diferentes de 5 instrumentos: viola, flauta travesera, violonchelo, saxo tenor y corno inglés. Todas las mezclas siguen la misma estructura: Instrumento A (3s) | Instrumento B (3s) | Mezcla de Instrumento A + Instrumento B (3s). Dicha estructura se aprecia en la Figura 4.1.

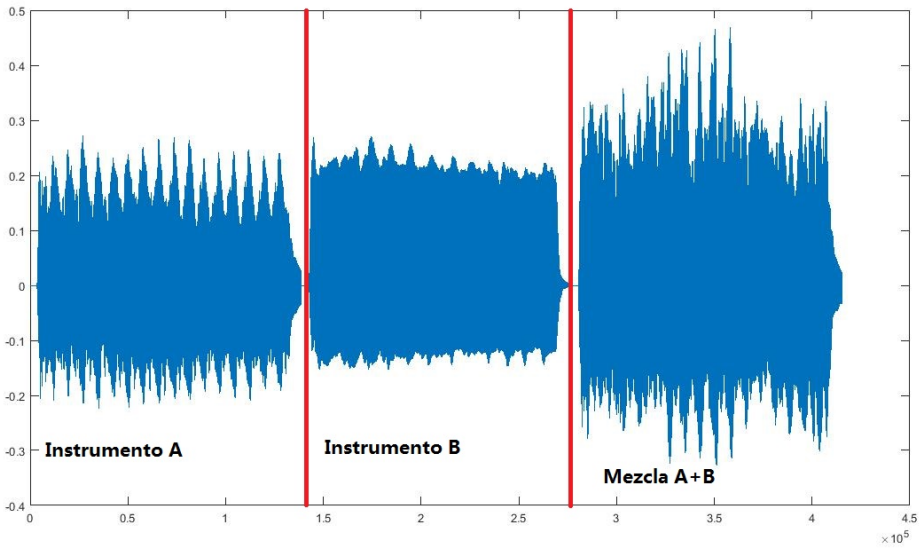

Figura 4.1 Estructura de las pistas de audio de entrada del algoritmo.

Todas las pistas de audio se han generado mediante un software de edición de audio[1], renderizando la nota C4 (Do central) correspondiente a los 261.63 Hz. Los datos se han codificado en archivos WAV a 44.1 kHz y 16 bits. Todas las pistas son mono canal.

[1] VIENNA SYMPHONIC LIBRARY (*https://vsl.co.at*)

4.2 Algoritmo paso a paso

1. Para empezar, es esencial pasar las pistas de audio a matrices de datos para poder trabajar con ellos. De esto se encarga la función *audioread*.

2. Una vez obtenida la matriz de datos de entrada, se calcula su STFT mediante la función *spectrogram*. La STFT se calcula con una ventana hamming de 1024 muestras de longitud, un solape del 50 % y una NFFT de 1024 puntos, siendo estos los puntos que se usan para calcular la Transformada Discreta de Fourier (DFT), necesaria para el cálculo de la STFT, como se explica en la Sección 3.2.1.

3. Con la STFT calculada y guardada en una variable a la que hemos llamado XSTFT, llamamos a la función CFM, que ha de ser creada anteriormente. Esta función recibe la STFT de los datos de entrada, la frecuencia de muestreo que devuelve la función *audioread* y los valores de α y β, y devuelve la SAR (Relación Señal a Artefacto), SDR (Relación Señal a Distorsión) y SIR (Relación Señal a Interferencia).

   A partir de este punto, se detalla paso a paso lo que realiza la función CFM.

4. Lo primero que hace la función, es calcular el número de filas y columnas que tendrá la matriz X, que corresponden a las filas y columnas de XSTFT. También se inicializan las variables que marcan el número de filas y columnas que tendrá cada parche, en nuestro caso $N_a = 4$ y $N_b = 64$ respectivamente. Por último, se calcula el número de parches que tendrá cada fila y cada columna y se guarda en las variables N_f y N_t.

5. Se calcula el tensor G, para ello se anidan dos bucles *for* para recorrer la matriz XSTFT e ir guardando en G parches de 4×64, teniendo en cuenta que los parches tienen un 50 % de solape, se obtiene un tensor de dimensión $4 \times 64 \times 256 \times 25$.

6. Una vez obtenido el tensor G, se calcula la matriz X realizando la 2D-DFT a cada parche de G. A continuación y siguiendo el Algoritmo 1, se inicializa V como el valor absoluto de X elevado a α, esto se ha hecho para simplificar los siguientes bucles iterativos.

7. Siguiendo lo dictado por el Algoritmo 1, tenemos que inicializar el tensor $P_a j$ (en el algoritmo equivale a $\hat{P}^{\alpha}$). Este tensor es de tamaño $N_a \times N_b \times N_f \times N_t \times J$ donde J es una variable que almacena el número de fuentes, en nuestro caso $J = 2$, la función de este tensor se explica en la Sección 3.2.4 y no es más que una variable de un modelo de factorización para calcular las densidades de modulación de las fuentes. Para poder dar valores al tensor, primero tenemos que inicializar de forma aleatoria, con la función *randn*, el tensor A (en el algoritmo $A_j(a,b,f)$), de tamaño $N_a \times N_b \times N_f \times J$ y la matriz H (en el algoritmo $H_j(t)$). Una vez hecho esto, anidamos dos bucles *for*, el primero se recorre tantas veces como fuentes tengamos y el segundo se recorre N_t veces. En cada iteración se realiza el producto de A por una entrada de H y se almacena en una entrada de $P_a j$.

8. En este paso, se calcula la densidad de modulación, que aparece en la Ecuación (3.1) como P^{α}, en nuestro algoritmo se almacenan en la variable P_a. Para este cálculo se ejecuta el Algoritmo 1, lo que significa que estamos ajustando los parámetros NMF del Modelo de Destino Recurrente, esto puede hacerse en este paso ya que previamente hemos inicializado todas las variables. El algoritmo se ejecuta dentro de un bucle iterativo, el cual se itera 100 veces, esta condición de parada es la recomendada por [53]. Una vez se llegue a la condición de parada, se toman los valores de la última actualización de $P_a j$ y se guardan en el tensor P_a como la suma de las $P_a j$ para ambas fuentes, teniendo así P_a una dimensión menos que $P_a j$.

   Llegados a este punto, podemos decir que hemos acabado con el modelo de factorización y a partir de ahora empezaremos con el proceso propio de separación de fuentes.

9. Para la separación de fuentes lo que tenemos que hacer es implementar la Ecuación (3.2) en Matlab®. Los resultados los guardaremos en una variable llamada S.

10. Lo siguiente que queremos, es obtener las formas de ondas correspondientes a los datos que hemos obtenido tras la separación. Lo primero que tendremos que hacer, será calcular la 2D-DFT inversa a cada parche de S que guardaremos en el tensor iS.

Una vez hecho eso, en un bucle tendremos que pasar el tensor *iS* de cuatro dimensiones, cuyos parches tienen un solape del 50 %, a una matriz de dos dimensiones sin solape. Esto se hace simplemente recorriendo el tensor *iS* con los valores adecuados y guardando los datos en el tensor *s*, que es un tensor porque contiene las matrices de las dos fuentes. Por último, tendremos que calcular la STFT inversa de los datos correspondientes a cada fuente, con los mismos parámetros que usamos en el punto 2 para el cálculo de la STFT y guardarlos en las variables x_1 y x_2 respectivamente. Estas variables, x_1 y x_2, podrían pasarse a pistas de audio con la función *audiowrite*. En la Figura 4.2 podemos ver un ejemplo para dos señales recuperadas correspondientes a una viola (Instrumento A) y a un saxo tenor (Instrumento B).

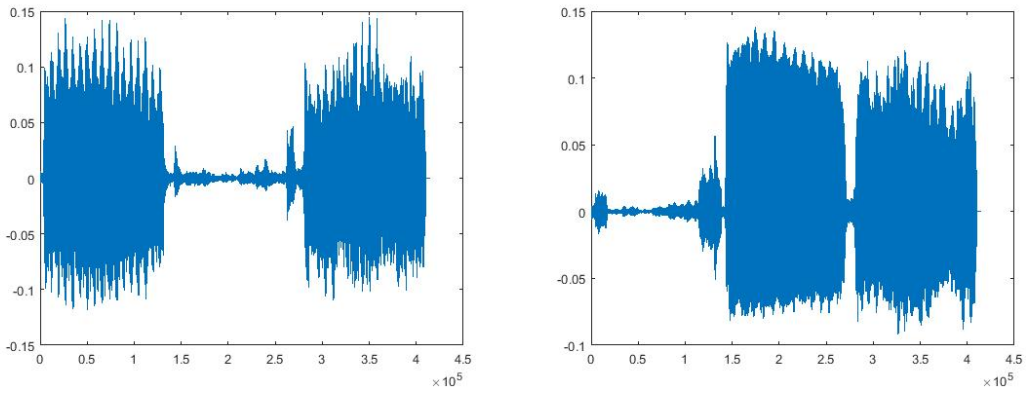

Figura 4.2 Viola y Saxo recuperados tras el proceso de separación de fuentes.

4.3 Evaluación de los resultados

Una vez hemos obtenido los resultados con el algoritmo expuesto en la sección anterior, es necesario medir la calidad de los resultados.

Existen diferentes medidas para evaluar la calidad de los resultados obtenidos, como la distorsión o la cantidad de señal original que se ha conseguido separar. En nuestro caso nos vamos a centrar en los métodos objetivos [57], en concreto en una serie de medidas denominadas como *medidas orientadas a la calidad de audio* (Audio Quality Oriented, AQO), por ser estas las más usadas de entre las medidas objetivas. En estos métodos, se supone que cada fuente estimada produce un modelo, en el que el error total cometido, se divide en tres términos relacionados con tres tipos de error, el modelo se expresa de la siguiente forma [56]:

$$\hat{s}(t) = s_{obj}(t) + e_{interf}(t) + e_{artef}(t) \tag{4.1}$$

donde $s_{obj}(t)$ es una deformación permitida de la fuente objetivo $s_i(t)$, e_{interf} representa la interferencia que ejercen las fuentes no deseadas y e_{artef} es el error generado en la propia separación. En otros casos, también habría que contar con otro error denominado e_{ruido}, que considera el ruido acústico, nosotros no lo tendremos en cuenta, debido a que las pistas de audio se han generado de forma sintética.

A partir de este modelo de distorsión, se definen las siguientes medidas para evaluar la separación de las fuentes:

- **SDR** (Signal to Distortion Ratio): compara las fuentes estimadas con las originales (error total), por lo que es la medida más usada para determinar la calidad de la separación de forma global.

$$SDR := 10\log_{10} \frac{\| s_{obj} \|^2}{\| e_{interf} + e_{artef} \|^2} \tag{4.2}$$

- **SIR** (Signal to Interference Ratio): mide la distorsión relativa causada por la interferencia de otras fuentes sobre la fuente objetivo.

$$SIR := 10\log_{10} \frac{\| s_{obj} \|^2}{\| e_{interf} \|^2} \tag{4.3}$$

- **SAR** (Signal to Artifacts Ratio): mide la distorsión relativa generada por el algoritmo al realizar la separación.

$$SAR := 10\log_{10} \frac{\| s_{obj} + e_{interf} \|^2}{\| e_{artef} \|^2} \tag{4.4}$$

4.4 Simulación 1

Esta primera simulación, se ha hecho usando las 10 mezclas de audio comentadas al principio del capítulo y ejecutando el algoritmo 5 veces para cada mezcla, como se hace en [53]. De estas ejecuciones se han obtenido 10 valores de SAR, SDR Y SIR para cada mezcla, formándose así una matriz de tamaño 100×3, donde cada columna corresponde a una medida y cada fila a un valor. En esta primera simulación, todas las ejecuciones se han hecho fijando los parámetros α y β al valor 1, al igual que se hace en [53].

Para obtener estos valores hemos usado el toolbox de Matlab® llamado *BSS Eval*, que se ha convertido en el estándar para medir la eficiencia de los algoritmos de BSS. Este toolbox fue presentado en [56] y nos hemos servido de [22][2] para poder ejecutar de forma correcta sus funciones. Entre las diversas funciones que tiene este toolbox, en este trabajo se ha usado la función *bss_eval_sources*, que según [22], sirve para evaluar las señales estimadas de fuentes monocanal. Esta función recibe dos matrices, una con las fuentes estimadas y otra con las fuentes originales y devuelve 4 vectores de tamaño № de fuentes $\times 1$ correspondientes a la SAR, SDR, SIR, y por último un vector que indica a qué fuente j original corresponde la fuente j estimada.

En la Figura 4.3 podemos apreciar un diagrama de cajas orientativo de los resultados obtenidos tras la ejecución de nuestro algoritmo. Este diagrama de cajas se obtiene ejecutando la función *boxplot* en Matlab®, función que recibe la matriz que contiene todos los valores calculados de SAR, SDR y SIR.

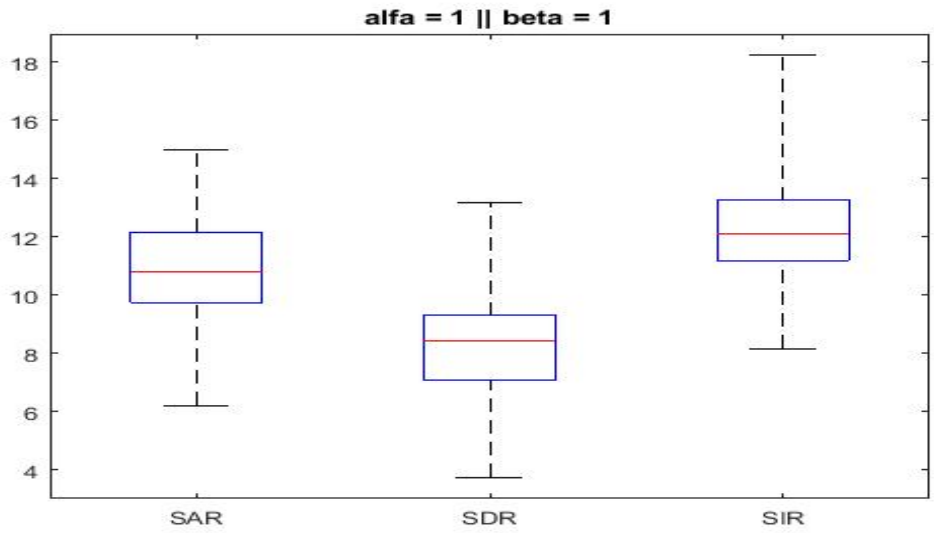

Figura 4.3 Diagrama de cajas que contiene 100 valores de las SAR, SDR y SIR de las 10 mezclas de audio para $\alpha = 1$ y $\beta = 1$.

[2] *http://bass-db.gforge.inria.fr/bss_eval/*

Tabla 4.1 Valores correspondientes al diagrama de cajas de la Figura 4.3.

	SAR	SDR	SIR
Máximo	14,95	13,16	18,05
Media	10,84	8,42	12,05
Mínimo	6,2	3,74	8,18

Para poder evaluar con una referencia los resultados obtenidos, hemos representado los expuestos en [53] en la Figura 4.4. Gracias a que el señor *Fabian-Robert Stöter* nos ofreció las pistas de audio resultante de sus simulaciones en [53], se ha podido comparar estas pistas con las de los instrumentos usando de nuevo *BSS Eval*.

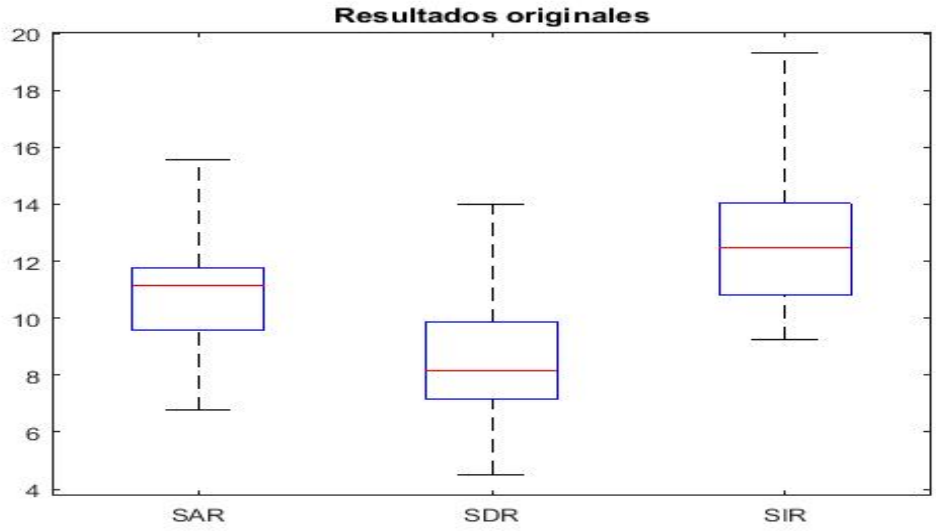

Figura 4.4 Diagrama de cajas que contiene 100 valores de las SAR, SDR y SIR de las 10 mezclas de audio para $\alpha = 1$ y $\beta = 1$.

Tabla 4.2 Valores correspondientes al diagrama de cajas de la Figura 4.4.

	SAR	SDR	SIR
Máximo	15,54	14	19,31
Media	11,16	8,16	12,45
Mínimo	6,78	4,5	9,24

Los resultados son satisfactorios ya que se asemejan considerablemente, e incluso en nuestra simulación, la media de la SDR es un poco mejor. Podemos afirmar, que el modelo presentado en [53] se ha implementado de forma correcta en Matlab®. En ambos resultados aparece una notable dispersión de los valores tanto en SAR, SDR como en SIR, hecho que se razonará en las próximas simulaciones ya que no se comenta en [53].

4.5 Simulación 2: Estudio de las Alfa-Beta divergencias

En esta sección vamos a explicar cómo se ha ejecutado en Matlab® el estudio sobre las AB divergencias expuesto en la Sección 3.3.

- Se lee la pista de audio con la función *audioread*.
- Se calcula la STFT de la pista leída en el punto anterior con la función *spectrogram*, se han usado los mismos parámetros que en la Sección 4.2, es decir, una ventana hamming de 1024 muestras de longitud, un solape del 50 % y una NFFT de 1024 puntos. Se guarda en la variable XSTFT.
- Se anidan 3 bucles *for*, uno para alfa, otro para beta y otro para repetir 5 veces la llamada a la función CFM, que recibe la variable XSTFT, la frecuencia de muestreo que devuelve la función *audioread* y los valores de alfa y beta.

  Para este estudio hemos tomado valores de alfa y beta desde -2 a 2 con un paso de 0.1. Cada vez que se sale del bucle que llama a la función CFM se guardan los valores de la SAR, SDR y SIR.
- Una vez terminado el bucle, que en la realidad se ha hecho por bloques, se tienen 3 matrices de tamaño 41×41 que contienen los valores de la SAR, SDR, y SIR correspondiente a cada valor de alfa y beta. Para representar estas matrices de forma gráfica de manera que sean fácilmente interpretables se ha usado la función *image* (ver Figura 4.5), escalando la gama de colores para cada figura.

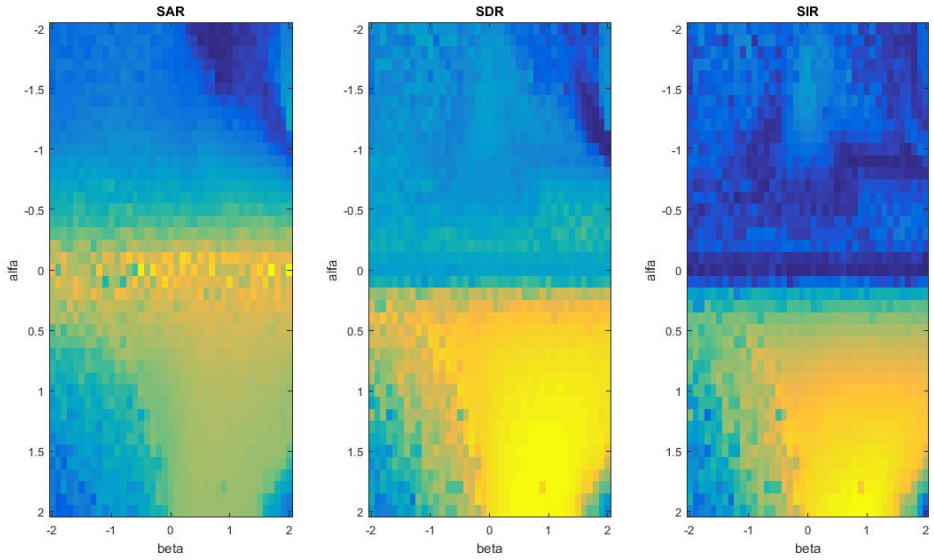

Figura 4.5 Representación gráfica de los valores de la SAR, SDR y SIR en función de los parámetros α y β para la mezcla de viola y saxo tenor.

4.5.1 Análisis de los resultados

Realizado el estudio de todas las mezclas, obtenemos la combinación óptima de los parámetros que maximiza el valor de SDR para cada una de ellas. En la Figura 4.6, podemos ver los diferentes screening de cada mezcla y en la Figura 4.7, el correspondiente a la mezcla de viola y saxo tenor para ver un screening con mayor resolución. La correspondencia de los acrónimos es la siguiente: C $\equiv$ violonchelo; CI $\equiv$ corno inglés; F $\equiv$ flauta; ST $\equiv$ saxo tenor y V $\equiv$ violín.

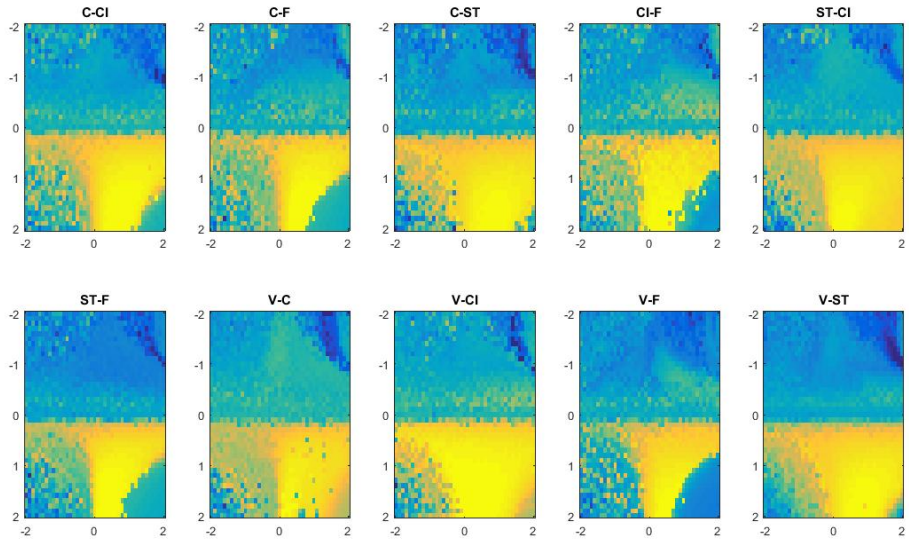

Figura 4.6 Representación gráfica de los valores de la SDR en función de los parámetros α y β para las 10 mezclas.

Tabla 4.3 Valores óptimos de los parámetros α y β para cada mezcla y valor SDR máximo.

	C-CI	C-F	C-ST	CI-F	ST-CI	ST-F	V-C	V-CI	V-F	V-ST
α	1,8	1,6	2	2	2	1,6	1,9	2	2	1,8
β	0,6	0,7	0,7	0,4	0,3	0,4	0,3	0,5	0,6	0,8
SDR	7,87	11,62	9,27	8,3	8,29	10,53	8,47	8,8	7,24	9,26

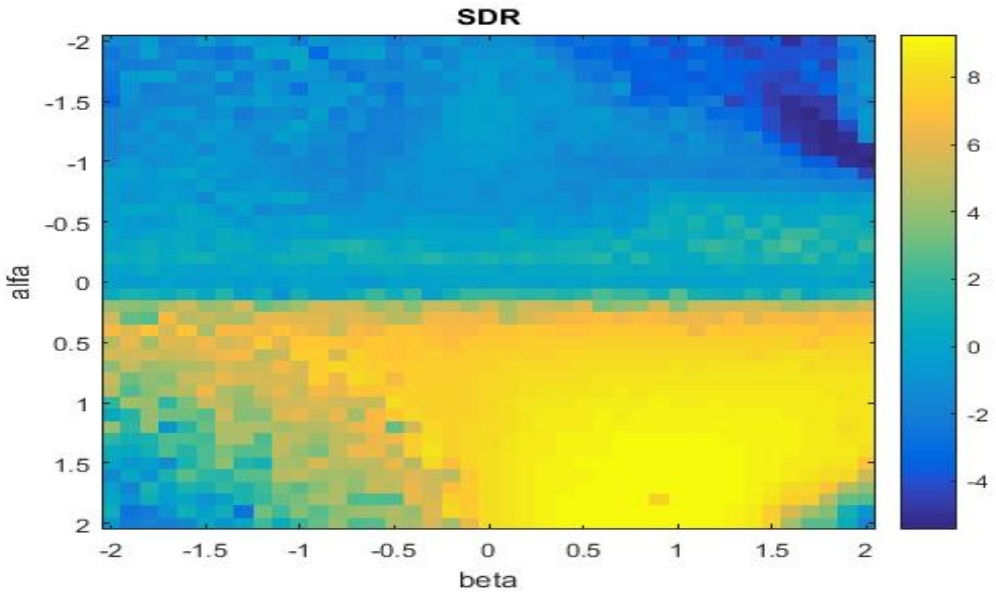

Figura 4.7 Representación gráfica de los valores de la SDR en función de los parámetros α y β para la mezcla de viola y saxo tenor.

4.6 Simulación 3

Tras el estudio de las AB-divergencias, se ha realizado una simulación parecida a la Simulación 1 (Sección 4.4), pero en este caso, usando los parámetros óptimos para cada mezcla.

En la Figura 4.8 se muestra un diagrama de cajas con 100 valores de SAR, SDR y SIR; 10 valores de cada mezcla.

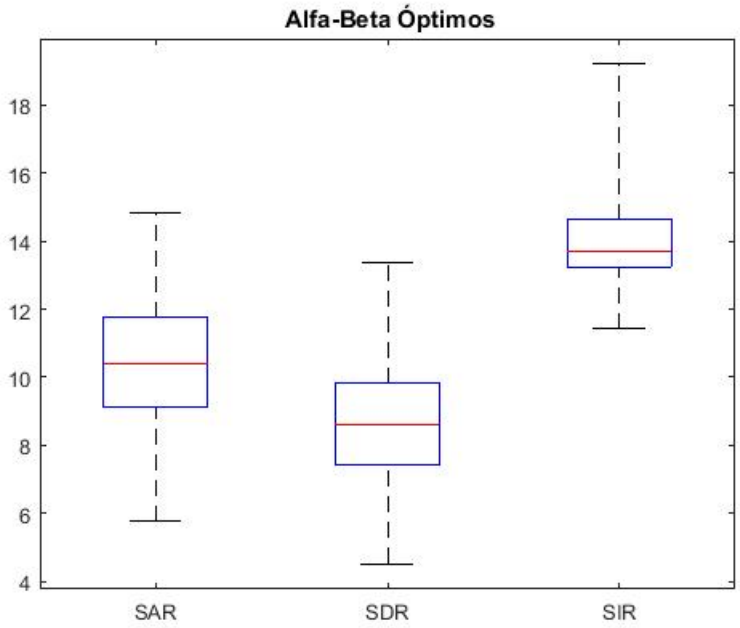

Figura 4.8 Diagrama de cajas que contiene 100 valores de las SAR, SDR y SIR de las 10 mezclas de audio para α y β óptimos de cada mezcla.

Tabla 4.4 Valores correspondientes al diagrama de cajas de la Figura 4.8.

	SAR	SDR	SIR
Máximo	14,83	13,35	19,22
Media	10,41	8,62	13,69
Mínimo	5,78	4,5	11,45

Se aprecia una mejora con respecto a la primera simulación (Sección 4.4), en concreto, la media de la SDR ha mejorado en 0,2 puntos para nuestros resultados y en 0,46 para los resultados originales. Los valores obtenidos para la SDR en la Figura 4.8, se han representado también para cada mezcla en la Figura 4.9.

En la Figura 4.9, hemos observado que dentro de las cajas de una misma mezcla, hay valores considerablemente dispares, hecho que se puede asegurar viendo la Tabla 4.5. Por ello, hemos calculado la desviación típica de cada mezcla, y se ha representado en la Tabla 4.6.

En primer lugar, y para entender bien los resultados obtenidos y su dispersión, es necesario conocer que dentro de los 10 valores que se han obtenido para cada mezcla, 5 corresponden a la separación del Instrumento A y las otras 5, al Instrumento B (ver Figura 4.1). En la Tabla 4.5, hemos sombreado las filas pares, para así ver con mayor facilidad que los valores de las filas sombreadas corresponden a un instrumento y las no sombreadas al otro.

Observando la Tabla 4.6, podemos ver que hay mezclas con una dispersión baja, coma la mezcla entre saxo tenor y corno inglés y otras con una desviación típica considerablemente alta, como la formada por viola y

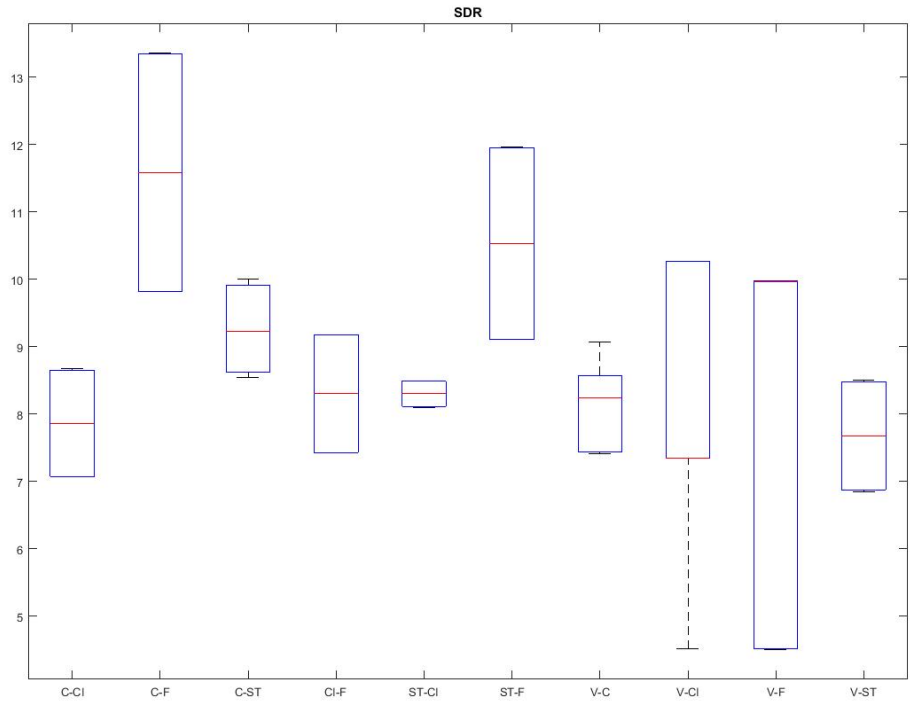

Figura 4.9 Diagrama de cajas que contiene 10 valores de SDR de cada mezcla.

Tabla 4.5 Valores correspondientes al diagrama de cajas de la Figura 4.9.

C-CI	C-F	C-ST	CI-F	ST-CI	ST-F	V-C	V-CI	V-F	V-ST
7,07	9,82	8,61	7,42	8,1	9,12	7,42	7,34	9,96	6,84
8,66	13,35	9,94	9,18	8,49	11,94	8,54	10,26	4,52	8,5
7,07	9,82	8,54	7,44	8,14	9,11	7,41	7,34	9,96	8,88
8,65	13,35	9,99	9,17	8,45	11,95	8,57	10,26	4,5	8,47
7,07	9,82	8,65	7,43	8,09	9,1	7,87	7,34	9,98	6,87
8,66	13,35	9,84	9,17	8,49	11,95	9,07	10,26	4,51	8,48
7,07	9,82	8,66	7,43	8,11	9,11	7,43	7,34	9,96	6,89
8,67	13,35	9,8	9,17	8,48	11,95	8,54	10,26	4,51	8,46
7,07	9,82	8,63	7,42	8,17	9,14	7,92	7,34	9,97	6,87
8,64	13,34	9,9	9,18	8,43	11,93	9,06	4,51	9,97	8,47

Tabla 4.6 Desviación típica de los valores de la Tabla 4.5 .

C-CI	C-F	C-ST	CI-F	ST-CI	ST-F	V-C	V-CI	V-F	V-ST
0,83	1,86	0,67	0,92	0,18	1,49	0,65	1,95	2,82	0,85

flauta. De estos datos, podemos interpretar que el algoritmo es dependiente del instrumento que se quiere separar, ya que en una misma mezcla, es capaz de separar de forma más eficiente a uno de los instrumentos.

También se aprecia que, para un mismo instrumento, el algoritmo en ocasiones obtiene un valor notablemente menor que la media (última entrada de la columna correspondiente a la mezcla V-CI de la Tabla 4.6) y en otras ocasiones mayor (última entrada de la columna correspondiente a la mezcla V-F de la Tabla 4.6).

4.7 Simulación 4

Una vez comprobada la dependencia del algoritmo a los instrumentos, se ha realizado una simulación para estudiar la influencia de los parámetros α y β. Para ello, se ha ejecutado una simulación para las 10 mezclas pero, en este caso, con los valores óptimos obtenidos en la Simulación 2 (Sección 4.5) para la mezcla de violonchelo y saxo tenor. En la Figura 4.10, se muestra un diagrama de cajas con los resultados obtenidos.

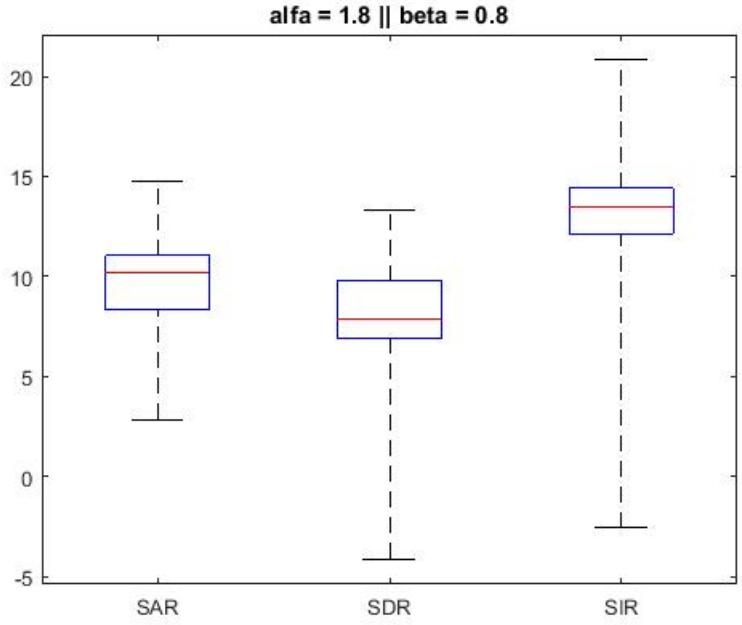

Figura 4.10 Diagrama de cajas que contiene 100 valores, 10 de cada mezcla para $\alpha = 1.8$ y $\beta = 0.8$.

Tabla 4.7 Valores correspondientes al diagrama de cajas de la Figura 4.10.

	SAR	SDR	SIR
Máximo	14,76	13,35	20,83
Media	10,2	7,84	13,47
Mínimo	2,84	-4,14	-2,54

Se aprecia en la Figura 4.10, que los valores de alfa y beta que consideramos óptimos para la mezcla de violonchelo y saxo tenor, no lo son para las demás mezclas. Podemos ver en la Tabla 4.7, que hay valores negativos en SDR y en SIR, algo significativo de una muy mala separación. Comparando con los resultados obtenidos en la Simulación 3 (Sección 4.6), la media de la SDR ha bajado 0.78, por lo que podemos afirmar que la eficiencia del algoritmo también depende de los valores de los parámetros α y β.

Debido a los valores tan dispares que se aprecian en la Tabla 4.7, se han representado los valores de SDR para cada mezcla en la Figura 4.11 y en la Tabla 4.8.

Según los valores de la desviación típica expuestos en la Tabla 4.9, podemos afirmar, como ya se hiciera en la Simulación 3 (Sección 4.6), que el algoritmo es dependiente de los instrumentos. Mientras que en la mezcla de saxo tenor y corno inglés la desviación es muy baja, en otras como la de viola y violonchelo, es notablemente alta.

En la Figura 4.11, podemos distinguir como algunos valores están muy alejados de la media para los casos de mezclas entre corno inglés y flauta; viola y violonchelo; viola y flauta; y viola y saxo tenor. Este hecho aumenta la dispersión y puede deberse a que en alguna de las 10 ejecuciones que se ha hecho para esa mezcla el algoritmo no ha sido capaz de detectar las diferentes componentes o que simplemente, como se dijera en la simulación anterior, el algoritmo es más eficiente para uno de los instrumentos de la mezcla.

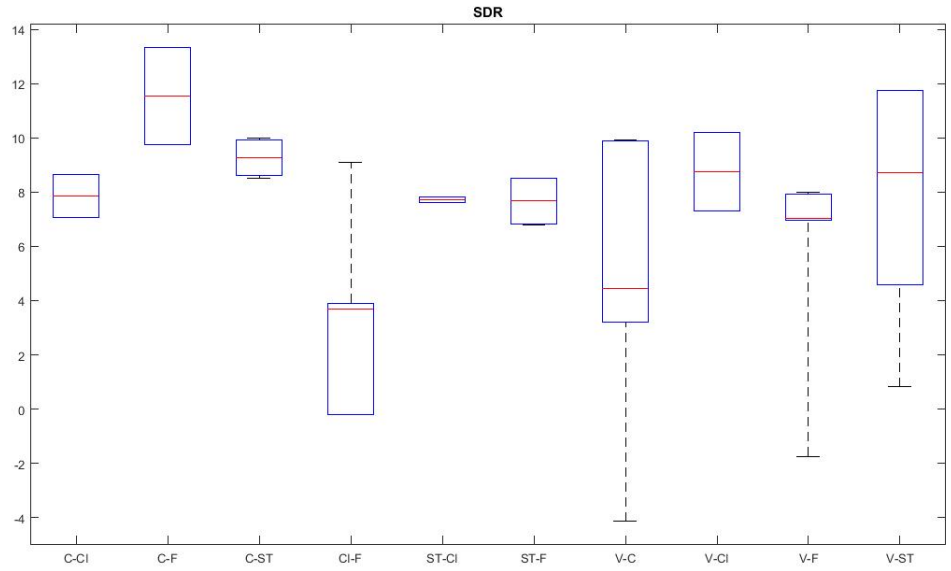

Figura 4.11 Diagrama de cajas donde cada caja representa 10 valores SDR de cada una de las 10 mezclas $\alpha = 1.8$ y $\beta = 0.8$.

Tabla 4.8 Valores correspondientes al diagrama de cajas de la Figura 4.11.

C-CI	C-F	C-ST	CI-F	ST-CI	ST-F	V-C	V-CI	V-F	V-ST
7,06	9,77	8,62	-0,22	7,61	6,8	-4,14	7,3	6,98	3,89
8,67	13,34	9,91	3,9	7,83	8,51	3,19	10,21	7,99	6,97
7,06	9,77	8,63	-0,22	7,61	6,81	4,46	7,3	7,03	8,74
8,66	13,34	9,89	3,9	7,84	8,52	9,91	10,21	7,86	11,76
7,05	9,77	8,61	-0,22	7,61	6,85	4,46	7,3	-1,75	0,82
8,65	13,34	9,92	3,9	7,83	8,5	9,91	10,22	-0,68	4,6
7,05	9,77	8,55	-0,14	7,61	6,88	-3,73	7,3	7,02	8,74
8,65	13,35	9,95	3,46	7,84	8,47	3,49	10,2	7,92	11,76
7,05	9,77	8,51	7,4	7,61	6,83	4,46	7,3	7,06	8,74
8,66	13,34	10	9,1	7,84	8,51	9,91	10,2	7,96	11,76

Tabla 4.9 Desviación típica de los valores de la Tabla 4.8 .

C-CI	C-F	C-ST	CI-F	ST-CI	ST-F	V-C	V-CI	V-F	V-ST
0.84	1.88	0.71	3.33	0.12	0.88	5.08	1.53	3.7	3.72

4.8 Simulación 5

En esta ocasión, lo que se ha hecho es cambiar la estructura de las pistas de audio que se le han pasado al algoritmo. Estas pistas, tienen una duración de 3 segundos y contienen solo la mezcla de dos instrumentos tocando la misma nota y ejecutando un vibrato, ver Figura 4.12.

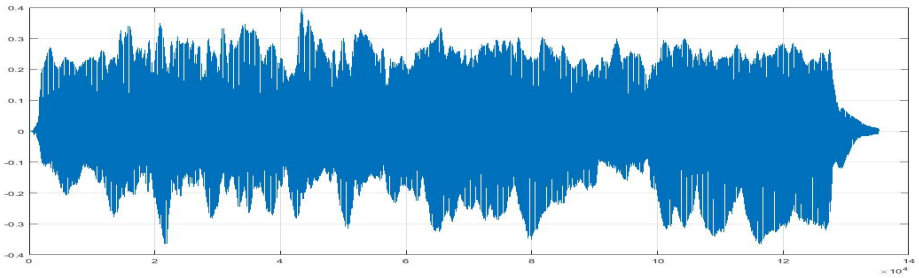

Figura 4.12 Estructura de las pistas de audio de 3 segundos.

Se han vuelto a usar las 10 mezclas anteriores, ahora recortadas a 3 segundos y los resultados de la separación se aprecian en la Figura 4.13. Para una óptima comparación, se han usado los mismos valores de los parámetros α y β que en la primera simulación.

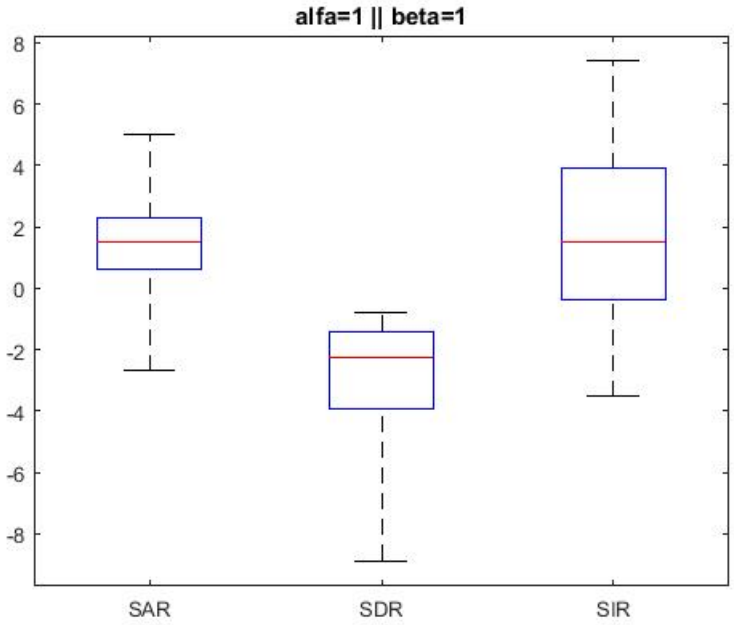

Figura 4.13 Diagrama de cajas que contiene 100 valores de las SAR, SDR y SIR de las 10 mezclas de audio de 3 segundos para $\alpha = 1$ y $\beta = 1$.

Tabla 4.10 Valores correspondientes al diagrama de cajas de la Figura 4.13.

	SAR	SDR	SIR
Máximo	5,01	-0,79	7,41
Media	1,51	-2,23	1,53
Mínimo	-2,66	-8,88	-3,49

Se aprecia en la Figura 4.13 y se corrobora con los valores de la Tabla 4.10, que la separación es muy mala. El 100 % de los 100 valores que hemos obtenido para la SDR son negativos.

Con esta simulación, se ha demostrado que el algoritmo realiza un proceso de aprendizaje cuando recibe los sonidos originales, como en la primera simulación, donde detecta claramente las diversas componentes y eso le sirve para separar. En este caso, el algoritmo es incapaz de detectar dichas componentes y por tanto, no ofrece una solución válida para la separación.

4.9 Simulación 6

Se ha probado el algoritmo con instrumentos diferentes a los usados en [53], también incluidos en el dataset aportado por *Fabian-Robert Stöter*. En este caso, se han elegido 4 instrumentos: contrabajo, clarinete, guitarra eléctrica y órgano. Las mezclas tienen la misma estructura que las usadas en la primera simulación y todos los instrumentos se encuentran tocando una nota C4 mientras ejecutan un vibrato.

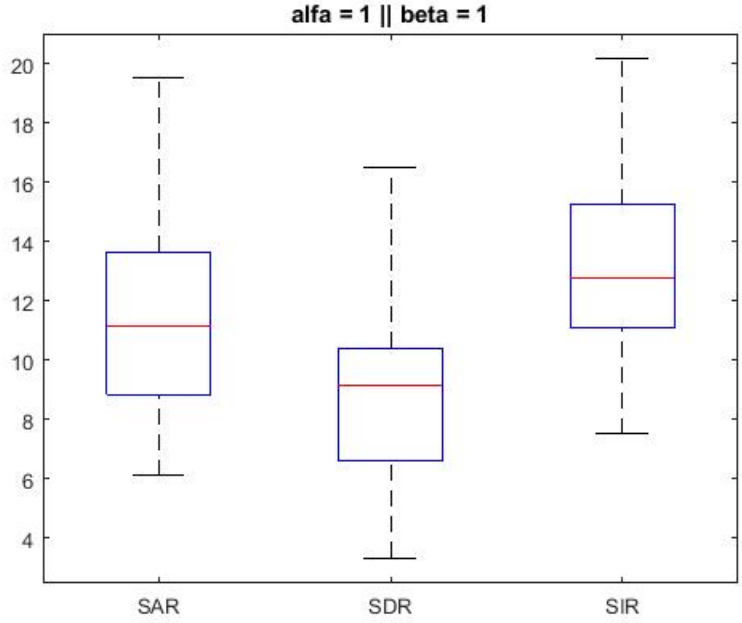

Figura 4.14 Diagrama de cajas que contiene 60 valores de las SAR, SDR y SIR de las 6 mezclas de audio para $\alpha = 1$ y $\beta = 1$.

Tabla 4.11 Valores correspondientes al diagrama de cajas de la Figura 4.14.

	SAR	SDR	SIR
Máximo	19,52	16,48	20,16
Media	11,17	9,12	12,79
Mínimo	6,1	3,31	7,51

Comparando con los valores de la primera simulación, se aprecia que hemos obtenido unos resultados ligeramente mejores. Centrándonos en los valores medios de la SDR, la media ha mejorado en 0,7 puntos. Se aprecia en este caso una mayor dispersión de los resultados frente a la Simulación 1 (Sección 4.4).

También se ha hecho una simulación para los valores óptimos de alfa y beta, calculados en la segunda simulación para la mezcla de violonchelo y saxo tenor, como se hiciera en la Simulación 4 (Sección 4.7). En este caso, si que se aprecia una mayor mejora de la SDR, con un aumento de 1,44 puntos frente a los valores expuestos en la Tabla 4.7 de la tercera simulación.

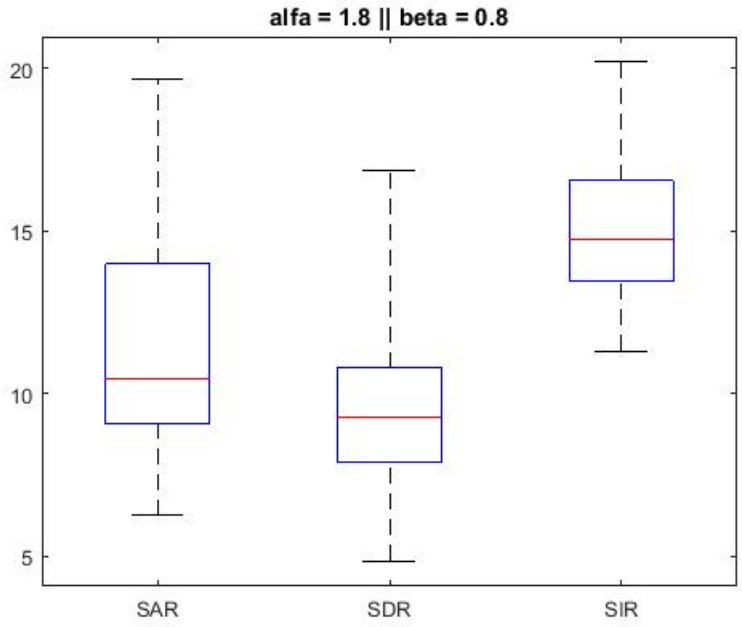

Figura 4.15 Diagrama de cajas que contiene 60 valores de las SAR, SDR y SIR de las 6 mezclas de audio para $\alpha = 1.8$ y $\beta = 0.8$.

Tabla 4.12 Valores correspondientes al diagrama de cajas de la Figura 4.15.

	SAR	SDR	SIR
Máximo	19,68	16,86	20,2
Media	10,44	9,28	14,73
Mínimo	6,25	4,84	11,31

Estos resultados, nos reafirman en el hecho de que el algoritmo es dependiente no solo de los instrumentos, si no que también lo es de los parámetros α y β.

4.10 Simulación 7

La última simulación realizada, se ha basado en ejecutar un algoritmo de separación NMF con las 10 mezclas utilizadas en la Simulación 1 (Sección 4.4). Así, podremos comparar el funcionamiento de CFM frente a NMF.

El algoritmo usado se ha tomado de [32][3]. Se ha ejecutado siguiendo la recomendación de los autores y una vez hecha la separación, se ha utilizado el toolbox *BSS Eval* para calcular la SAR, SDR y SIR. Se han hecho dos simulaciones para dos divergencias diferentes, la primera para Itakura-Saito y la segunda para Kullback-Leibler.

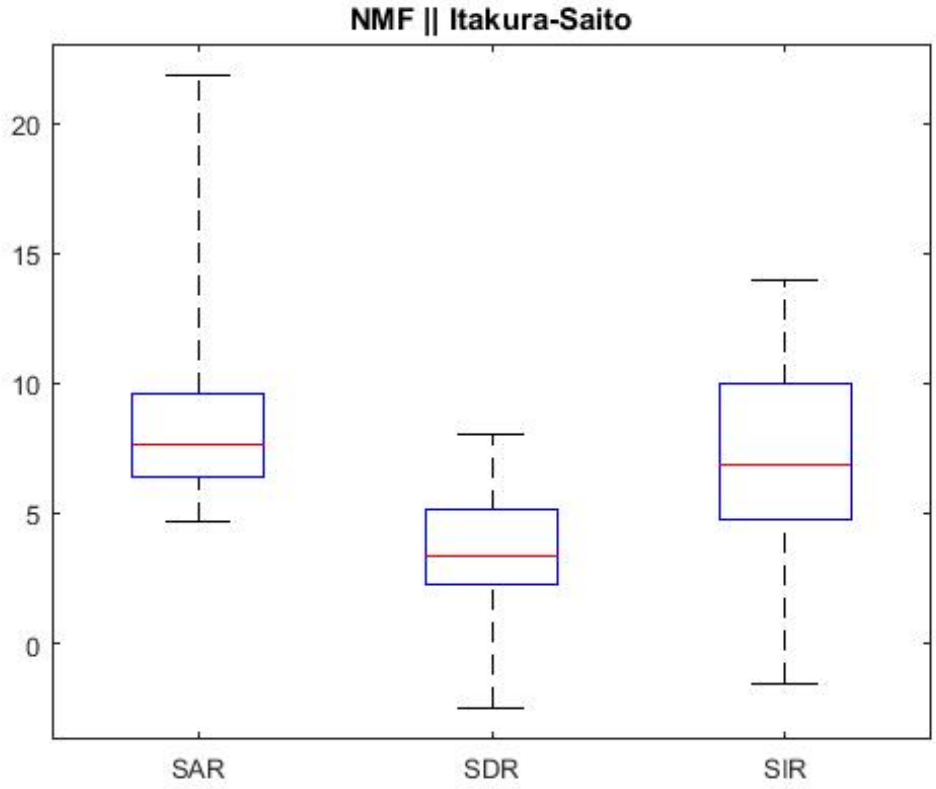

Figura 4.16 Diagrama de cajas que contiene 20 valores de las SAR, SDR y SIR de las 10 mezclas de audio para la divergencia de Itakura-Saito.

Tabla 4.13 Valores correspondientes al diagrama de cajas de la Figura 4.16.

	SAR	SDR	SIR
Máximo	21,85	8,03	14
Media	7,62	3,39	6,84
Mínimo	4,66	-2,5	-1,53

Frente a la primera simulación, la media de la SDR ha bajado 4,97 puntos para la divergencia Itakura-Saito y 5,06 puntos para Kullback-Leibler, mientras que para la simulación donde se han optimizado los parámetros α y β el descenso de la SDR ha sido de 5,23 y 5,32 respectivamente. Con estos resultados, queda demostrado que NMF no es un modelo eficaz para fuentes unísonas mono-canal moduladas en tiempo y frecuencia, tal como se ha explicado teóricamente en la introducción del Capítulo 3.

[3] *https://github.com/EliasKokkinis/audio-source-separation*

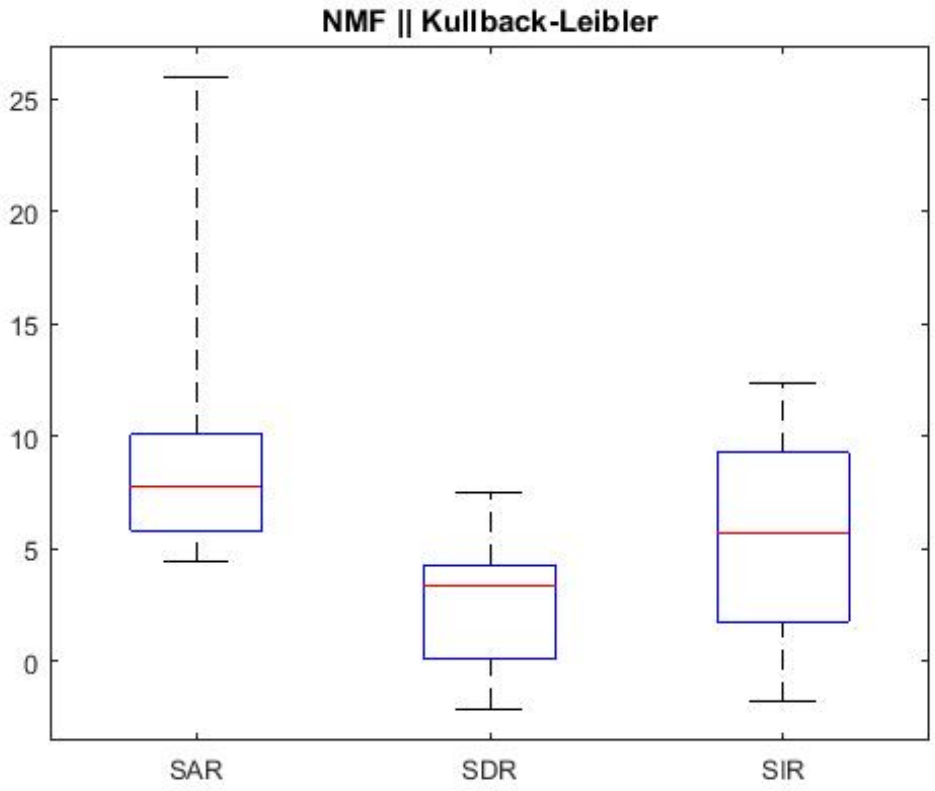

Figura 4.17 Diagrama de cajas que contiene 20 valores de las SAR, SDR y SIR de las 10 mezclas de audio para la divergencia de Kullback-Leibler.

Tabla 4.14 Valores correspondientes al diagrama de cajas de la Figura 4.17.

	SAR	SDR	SIR
Máximo	25,95	7,48	12,32
Media	7,76	3,3	5,67
Mínimo	4,39	-2,15	-1,84

Se aprecia en estas simulaciones una fuerte dispersión, que nos indica que NMF también es dependiente de los instrumentos. En cuanto a la dependencia a los parámetros α y β, los resultados varían considerablemente menos que en las simulaciones realizadas con CFM.

5 Conclusiones y Líneas Futuras

5.1 Trabajo realizado y conclusiones

En este proyecto hemos trabajado sobre un método para explotar texturas de modulación recurrentes para el problema de la Separación Ciega de Fuentes, basándonos durante todo el trabajo en el presentado por *F.-R. Stöter et al.* en [53].

Para dar sentido a todo nuestro trabajo, se ha partido de un estudio teórico general de BSS y se han ido desarrollando de manera más extensa algunas técnicas y soluciones existentes, como NMF y sus propiedades, o las divergencias más utilizadas en la estimación de los parámetros.

Tras la introducción teórica, se ha hecho un estudio teórico-práctico para obtener las fórmulas del algoritmo usado para resolver el problema se separación de fuentes. Para ello se ha introducido el concepto de divergencia AB, sus propiedades y sus ventajas respecto a otras divergencias conocidas. Finalmente en este estudio, se ha usado un algoritmo multiplicativo, basado en el estudio teórico anterior.

Finalmente, se ha implementado el algoritmo en Matlab ® y se han realizado diversas simulaciones, presentando los resultados obtenidos. En función de los resultados obtenidos y su análisis, se han ido extrayendo varias conclusiones que presentamos a continuación.

En primer lugar y como conclusión más importante, los valores de nuestros resultados para mezclas de dos instrumentos tocando la misma nota mientras ejecutan un vibrato, indican que este método funciona bien en este desafiante escenario. Lo que implica directamente, que la implementación del modelo en Matlab ® se ha realizado de forma correcta.

En segundo lugar, del estudio de las divergencias AB, concluimos que su uso ha aportado mucha versatilidad al algoritmo, ya que al tratarse de una familia de divergencias que engloban a otras conocidas, pueden modificarse de manera sencilla las fórmulas para la estimación de las fuentes. También nos brinda la posibilidad de realizar un screening para determinar las zonas o puntos del plano alfa/beta donde los resultados son más favorables, como se ha hecho en la Sección 4.5. Se ha demostrado que para diferentes mezclas los valores óptimos de alfa y beta son diferentes, por lo que el estudio de las divergencias AB es de gran ayuda.

Por último, se puede afirmar que con el estudio de las divergencias AB se han encontrado combinaciones de alfa/beta que mejoran los resultados de [53], donde solo se presentan resultados para $\alpha = 1$ y $\beta = 1$, por lo que podemos considerar la realización del trabajo y sus resultados como altamente satisfactorios.

5.2 Líneas futuras

Tras la realización de este trabajo, consideramos que una de las mayores restricciones del mismo, es el tipo de pistas de audio con el que hemos trabajado. Sin duda, una de las líneas futuras se basa en probar el modelo con otras pistas de audio, sintéticas con diferente estructura y también con grabaciones reales, para ver cómo actúa el modelo frente al ruido.

Por otra parte, sería de gran interés mejorar la eficiencia de nuestro código en Matlab®, especialmente para el estudio de las AB divergencias ya que ha sido uno de los puntos negativos de este trabajo.

Debido a los resultados obtenidos en la Simulación 5 (Sección 4.8), es necesario en un futuro estudiar a fondo el aprendizaje realizado por el algoritmo y cómo optimizar la separación cuando no se tienen las pistas originales previas a la mezcla.

También podría estudiarse la aplicación del modelo al problema de la separación del habla. Ya se han usado algoritmos basados en NMF para este problema [11]. En la separación del habla, también se suele encontrar audio modulado en frecuencia y en amplitud, como características propias del habla, ya que el hablante no suele usar el mismo tono ni el mismo volumen durante una conversación, algo que puede ayudar a un correcto funcionamiento del modelo. Dichas modulaciones serían seguramente más variables que las estudiadas en este trabajo (que son siempre de 5 Hz), hecho que puede resultar una dificultad añadida para el modelo, además, las pistas de audio que recibiría no tendrían la misma estructura que en este estudio, algo que la Simulación 5 (Sección 4.8) ha demostrado que influye de forma muy negativa en la separación. Es común en los algoritmos de separación del habla, que haya una primera fase de entrenamiento del algoritmo con audios del conjunto de datos del problema a resolver, esto podría añadirse o ampliarse al actual modelo.

Índice de Figuras

Índice de Tablas

Índice de algoritmos

Bibliografía

[1] A. Asaei, M. E. Davies, H. Bourlard, and V. Cevher, *Computational methods for structured sparse component analysis of convolutive speech mixtures*, 2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), March 2012, pp. 2425–2428.

[2] R. Badeau, B. David, and G. Richard, *High-resolution spectral analysis of mixtures of complex exponentials modulated by polynomials*, IEEE Transactions on Signal Processing **54** (2006), no. 4, 1341–1350.

[3] Roland Badeau, *Gaussian modeling of mixtures of non-stationary signals in the time-frequency domain (hr-nmf)*, IEEE Workshop on Applications of Signal Processing to Audio and Acoustics, oct 2011.

[4] Roland Badeau and A.Dremeau, *Variational bayesian em algorithm for modeling mixtures of non-stationary signals in the time-frequency domain (hr-nmf)*, ICASSP International Conference on Acoustics, Speech, and Signal Processing, may 2013, pp. 6171–6175.

[5] Roland Badeau and M.D. Plumbey, *Multichannel hr-nmf for modelling convolutive mixtures of non-stationary signals in the time-frequency domain*, IEEE Workshop on Applications of Signal Processing to Audio and Acoustics, oct 2013.

[6] Y. Bar-Ness, J. W. Carlin, and M. L. Steinberger, *Bootstrapping adaptive interference cancelers - Some practical limitations*, Globecom '82 - Global Telecommunications Conference, 1982, pp. 1251–1255.

[7] T Barker and Tuomas Virtanen, *Non-negative tensor factorisation of modulation spectrograms for monaural sound source separation*, Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH, Jan 2013, pp. 827–831.

[8] J. F. Cardoso, *Multidimensional independent component analysis*, Acoustics, Speech and Signal Processing, 1998. Proceedings of the 1998 IEEE International Conference on, vol. 4, May 1998, pp. 1941–1944 vol.4.

[9] Jean-François Cardoso, *Blind signal separation: statistical principles*, Proceedings Of The IEEE **9** (1998), no. 10, 2009–2025.

[10] Jean-Francois Cardoso, *The three easy routes to independent component analysis, contrasts and geometry*, In Proc. ICA 2001, 2001, pp. 1–6.

[11] Y. Chen, *Single channel blind source separation based on nmf and its application to speech enhancement*, 2017 IEEE 9th International Conference on Communication Software and Networks (ICCSN), May 2017, pp. 1066–1069.

[12] Andrzej Cichocki, Sergio Cruces, and Shun-ichi Amari, *Generalized alpha-beta divergences and their application to robust nonnegative matrix factorization*, Entropy **13** (2011), no. 1, 134–170.

[13] Andrzej Cichocki, Rafal Zdunek, Anh Huy Phan, and Shun-Ichi Amari, *Nonnegative matrix and tensor factorizations: Applications to exploratory multiway data analysis and blid source separation*, 1ª ed., Wiley, 2009.

[14] P. Common and C. Jutten, *Handbook of blind source separation: Independent component analysis and applications*, 1 ed., Elsevier, 2010.

[15] A. Deleforge, F. Forbes, and R. Horaud, *Variational em for binaural sound-source separation and localization*, 2013 IEEE International Conference on Acoustics, Speech and Signal Processing, May 2013, pp. 76–80.

[16] A. P. Dempster, N. M. Laird, and D. B. Rubin, *Maximum likelihood from incomplete data via the em algorithm*, JOURNAL OF THE ROYAL STATISTICAL SOCIETY, SERIES B **39** (1977), no. 1, 1–38.

[17] N. Q. K. Duong, E. Vincent, and R. Gribonval, *Under-determined reverberant audio source separation using a full-rank spatial covariance model*, IEEE Transactions on Audio, Speech, and Language Processing **18** (2010), no. 7, 1830–1840.

[18] Ngoc Q. K. Duong, Emmanuel Vincent, and Rémi Gribonval, *Spatial location priors for gaussian model based reverberant audio source separation*, EURASIP Journal on Advances in Signal Processing **2013** (2013), no. 1, 149.

[19] Slim Essid and Alexey Ozerov, *A tutorial on nonnegative matrix factorisation with applications to audiovisual content analysis*, ICME International Conference on Multimedia and Expo, 2014.

[20] Cédric Févotte, Nancy Bertin, and Jean-Louis Durrieu, *Nonnegative matrix factorization with the itakura-saito divergence: With application to music analysis*, Neural Compututation **21** (2009), no. 3, 793–830.

[21] D. FitzGerald, M. Cranitch, and E. Coyle, *On the use of the beta divergence for musical source separation*, IET Irish Signals and Systems Conference (ISSC 2009), June 2009, pp. 1–6.

[22] C. Févotte, R. Gribonval, and E. Vincent, *Bss_eval toolbox user guide*, 2005.

[23] S. Greenberg and B. E. D. Kingsbury, *The modulation spectrogram: in pursuit of an invariant representation of speech*, 1997 IEEE International Conference on Acoustics, Speech, and Signal Processing, vol. 3, Apr 1997, pp. 1647–1650 vol.3.

[24] E.A.P. Habets, S. Gannot, and I. Cohen, *Late reverberant spectral variance estimation based on a statistical model*, IEEE Signal Processing Letters **16** (2009), no. 9, 770–773.

[25] Richard A. Harshman, *Foundations of the parafac procedure: Models and conditions for an 'explanatory' multi-modal factor analysis*, UCLA Working Papers in Phonetics (1970), no. 16, 1 – 84.

[26] R. Hennequin, R. Badeau, and B. David, *Nmf with time-frequency activations to model nonstationary audio events*, IEEE Transactions on Audio, Speech, and Language Processing **19** (2011), no. 4, 744–753.

[27] P. S. Huang, S. D. Chen, P. Smaragdis, and M. Hasegawa-Johnson, *Singing-voice separation from monaural recordings using robust principal component analysis*, 2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), March 2012, pp. 57–60.

[28] Aapo Hyvärinen and Erkki Oja, *Independent component analysis: Algorithms and applications*, Neural Networks (2000), no. 13, 411–430.

[29] Fumitada Itakura and Shuzo Saito, *Analysis synthesis telephony based on the maximum likelihood method*, 1968, pp. 17–20.

[30] Brian E.D Kingsbury, Nelson Morgan, and Steven Greenberg, *Robust speech recognition using the modulation spectrogram*, Speech Communication **25** (1998), no. 1, 117 – 132.

[31] T. Kinnunen, K. Lee, and H. Li, *Dimension reduction of the modulation spectrogram for speaker verification*, Odyssey 2008: The Speaker and Language Recognition Workshop, Jan 2008.

[32] Elias Kokkinis, Alexandros Tsilfidis, and Michael Tzannes, *Audio source separation*, ECESCON 8, April 2015.

[33] M. Kowalski, E. Vincent, and R. Gribonval, *Beyond the narrowband approximation: Wideband convex methods for under-determined reverberant audio source separation*, IEEE Transactions on Audio, Speech, and Language Processing **18** (2010), no. 7, 1818–1829.

[34] Matthieu Kowalski and Bruno Torrésani, *Sparsity and persistence: Mixed norms provide simple signal models with dependent coefficients*, **3** (2009).

[35] S. Kullback and R. A. Leibler, *On information and sufficiency*, The Annals of Mathematical Statistics **22** (1951), no. 1, 79–86.

[36] D D Lee and H S Seung, *Algorithms for nonnegative matrix factorization*, NIPS'00 Proceedings of the 13th International Conference on Neural Information Processing Systems, 2000.

[37] Daniel D. Lee and H. Sebastian Seung, *Learning of the parts of objects by non-negative matrix factorization*, Nature (1999), no. 401, 788–791.

[38] A. Liutkus and R. Badeau, *Generalized wiener filtering with fractional power spectrograms*, 40th International Conference on Acoustics, Speech and Signal Processing (ICASSP), 04 2015.

[39] M. Markaki and Y. Stylianou, *Using modulation spectra for voice pathology detection and classification*, 2009 Annual International Conference of the IEEE Engineering in Medicine and Biology Society, Sept 2009, pp. 2514–2517.

[40] G. Mysore and M. Sahani, *Variational inference in non-negative factorial hidden markov models for efficient audio source separation*, 29th Int. Conf. Machine Learning, 2012, pp. 1887–1894.

[41] Ganesh R. Naik and Wenwu Wang, *Blind source separation: Advances in theory, algorithms and applications*, 1ª ed., Springer, 2014.

[42] A. Ozerov and C. Fevotte, *Multichannel nonnegative matrix factorization in convolutive mixtures for audio source separation*, IEEE Transactions on Audio, Speech, and Language Processing **18** (2010), no. 3, 550–563.

[43] A. Ozerov, C. Févotte, R. Blouet, and J. L. Durrieu, *Multichannel nonnegative tensor factorization with structured constraints for user-guided audio source separation*, 2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), May 2011, pp. 257–260.

[44] Alexey Ozerov, Emmanuel Vincent, and Frédéric Bimbot, *A general flexible framework for the handling of prior information in audio source separation*, IEEE Transactions on Audio, Speech and Signal Processing, 2012, pp. 1118–1133.

[45] Pentti Paatero and Unto Tapper, *Positive matrix factorization: A non-negative factor model with optimal utilization of error estimates of data values*, Environmetrics **5** (1994), 111–126.

[46] L. Parra and C. Alvino, *Geometric source separation: merging convolutive source separation with geometric beamforming*, Neural Networks for Signal Processing XI: Proceedings of the 2001 IEEE Signal Processing Society Workshop (IEEE Cat. No.01TH8584), 2001, pp. 273–282.

[47] P.Bofill and M.Zibulevsky, *Underdetermined blid source representations*, Signal Processing (2001), no. 81, 2353–2362.

[48] Roland Badeau P.Magron and B.David, *Phase recovery in nmf for audio source separation: an insightful benchmark*, ICASSP International Conference on Acoustics, Speech, and Signal Processing, apr 2015, pp. 81–85.

[49] Z. Rafii and B. Pardo, *Repeating pattern extraction technique (repet): A simple method for music/voice separation*, IEEE Transactions on Audio, Speech, and Language Processing **21** (2013), no. 1, 73–84.

[50] G. Samoradnitsky and M. Taqqu, *Stable non-gaussian random processes: stochastic models with infinite variance,*, 1ª ed., vol. 1, CRC Press, 1994.

[51] H. Sawada, S. Araki, R. Mukai, and S. Makino, *Grouping separated frequency components by estimating propagation model parameters in frequency-domain blind source separation*, IEEE Transactions on Audio, Speech, and Language Processing **15** (2007), no. 5, 1592–1604.

[52] P. Smaragdis, *Convolutive speech bases and their application to supervised speech separation*, IEEE Transactions on Audio, Speech, and Language Processing **15** (2007), no. 1, 1–12.

[53] Fabian-Robert Stöter, Antoine Liutkus, Roland Badeau, Bernd Edler, and Paul Magron, *Common Fate Model for Unison Source Separation*, 41st International Conference on Acoustics, Speech and Signal Processing (ICASSP) (Shanghai, China), Proceedings of the 41st International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2016.

[54] F. R. Stöter, S. Bayer, B. Edler, and P. Magron, *Unison source separation*, 17th International Conference on Digital Audio Effects, September 2014, pp. 235–241.

[55] E. Vincent, N. Bertin, R. Gribonval, and F. Bimbot, *From blind to guided audio source separation: How models and side information can improve the separation of sound*, IEEE Signal Processing Magazine **31** (2014), no. 3, 107–115.

[56] E. Vincent, R. Gribonval, and C. Fevotte, *Performance measurement in blind audio source separation*, IEEE Transactions on Audio, Speech, and Language Processing **14** (2006), no. 4, 1462–1469.

[57] Emmanuel Vincent, *Improved perceptual metrics for the evaluation of audio source separation*, Latent Variable Analysis and Signal Separation (Berlin, Heidelberg) (Fabian Theis, Andrzej Cichocki, Arie Yeredor, and Michael Zibulevsky, eds.), Springer Berlin Heidelberg, 2012, pp. 430–437.

[58] Emmanuel Vincent, Shoko Araki, Fabian Theis, Guido Nolte, Pau Bofill, Hiroshi Sawada, Alexey Ozerov, Vikrham Gowreesunker, Dominik Lutter, and Ngoc Q. K. Duong, *The signal separation evaluation campaign (2007-2010): Achievements and remaining challenges*, Signal Process. **92** (2012), no. 8, 1928–1936.

[59] T. Virtanen, *Monaural sound source separation by nonnegative matrix factorization with temporal continuity and sparseness criteria*, IEEE Transactions on Audio, Speech, and Language Processing **15** (2007), no. 3, 1066–1074.

[60] Xinling Wen, *Research and simulation of linear instantaneous blind signal separation algorithm*, Advances in Computer Science, Environment, Ecoinformatics, and Education (Berlin, Heidelberg) (Song Lin and Xiong Huang, eds.), Springer Berlin Heidelberg, 2011, pp. 119–124.

[61] O. Yilmaz and S. Rickard, *Blind separation of speech mixtures via time-frequency masking*, IEEE Transactions on Signal Processing **52** (2004), no. 7, 1830–1847.

[62] G. Zhou, Q. Zhao, Y. Zhang, T. Adalı, S. Xie, and A. Cichocki, *Linked component analysis from matrices to high-order tensors: Applications to biomedical data*, Proceedings of the IEEE **104** (2016), no. 2, 310–331.