

DPTO. INGENIERÍA ELECTRÓNICA
ÁREA DE TEORÍA DE LA SEÑAL Y COMUNICACIONES
ESCUELA SUPERIOR DE INGENIEROS
INGENIERO DE TELECOMUNICACIÓN
UNIVERSIDAD DE SEVILLA

PROYECTO FIN DE CARRERA

**IMPLEMENTACIÓN DE UN ALGORITMO PARA
LA SEPARACIÓN CIEGA DE FUENTES CON
PEQUEÑOS RETARDOS**

AUTOR: JUAN ANTONIO LEÑERO BARDALLO

TUTOR: DR. SERGIO A. CRUCES ÁLVAREZ

SEPTIEMBRE DE 2005

PROYECTO FIN DE CARRERA

TÍTULO: Implementación de un algoritmo para la separación ciega de fuentes con pequeños retardos

TRIBUNAL CALIFICADOR:

PRESIDENTE:

VOCAL:

SECRETARIO:

CALIFICACIÓN:

FECHA DE LECTURA:

A mis padres

Resumen

El modelo clásico de ICA para la separación ciega de fuentes (BSS) considera que las mezclas de fuentes recibidas en los receptores se producen de forma instantánea. Esta suposición, aunque en la mayoría de los casos es aceptable, no es del todo cierta, ya que las señales siempre sufrirán un retardo en su propagación hacia los receptores.

En el caso de que las mezclas recibidas en los receptores sean de señales de voz, no es correcto considerar que no se producen retardos en la propagación de las fuentes porque éstas se propagarán a la velocidad del sonido y un mínimo desplazamiento entre los receptores, las fuentes o ambos hará que se produzcan retrasos apreciables. Para la separación de señales de voz, es más correcto emplear un modelo ICA que tenga en cuenta los retardos que se producen.

En el presente proyecto, se pretende abordar el estudio del comportamiento de un algoritmo para la separación de fuentes recibidas con pequeños retardos. Veremos que cuando los retardos son pequeños, es posible tener en cuenta su efecto sin necesidad de utilizar el modelo ICA convolutivo, de gran complejidad matemática, que considera el efecto de la respuesta impulsiva del canal.

Para la resolución del problema de separación teniendo en cuenta los retardos, se realizará un estudio un modelo ICA que considera que las mezclas se producen de forma instantánea. El modelo también considera que dichas mezclas están formadas por las fuentes de interés y sus derivadas primeras. Así se tiene en cuenta el efecto de los retardos. Para que el modelo sea válido, los retardos introducidos deben estar acotados, ya que de lo contrario, las hipótesis en las que se basa el algoritmo no se cumplirán.

Además, en el modelo analizado, se introduce ruido para modelar de forma más precisa las mezclas de señales que se recibirían en una situación real.

En definitiva, se propone el estudio de un método de separación que, como novedad, considera el efecto de los retardos de las fuentes en su propagación y el efecto del ruido existente en el medio.

AGRADECIMIENTOS

Llegados a este punto, me gustaría hacer constar mi más profunda gratitud a mi tutor Sergio Cruces por la inestimable ayuda prestada para la elaboración del presente documento. He de agradecerle sus valiosos consejos, su total disponibilidad siempre que he necesitado su ayuda y su interés por la consecución de los objetivos marcados en el proyecto.

También me gustaría agradecer a mis padres y hermanos el apoyo y comprensión prestado durante mis años de estudiante. Sin ellos, la realización de este proyecto no hubiera sido posible.

Por último, me gustaría hacer referencia a mis compañeros de carrera durante estos últimos cinco años. Gracias a ellos, mi estancia en la universidad me ha resultado francamente grata y creo que nunca olvidaré los buenos momentos que hemos pasado juntos.

NOTACIÓN

A continuación, se muestran los símbolos más usados:

$\ \cdot \ , \cdot $	Operadores norma y módulo
$(\cdot)^H, (\cdot)^T$	Operadores hermítico y de transposición
$(\cdot)^*$	Operador de conjugación
δ_{ij}	Función de Kronecker
$\delta(t)$	Función impulso unitario
λ	Autovalores de una matriz
σ^2	Varianza de la señal de ruido en los receptores
τ_{ij}	Retardo desde la fuente j al receptor i
c	Velocidad de propagación de las fuentes en el medio
c_{ij}	Atenuación desde la fuente j hasta el receptor i
f_{max}, f_{min}	Frecuencias máxima y mínima de las fuentes transmitidas
$h(t)$	Filtro auxiliar utilizado para obtener la matriz de autocovarianzas filtrada
$\mathbf{n}, \mathbf{n}(t)$	Vector de ruido recibido en los receptores
$pinv(\cdot)$	Matriz pseudoinversa
$\mathbf{s}, \mathbf{s}(t)$	Vector de fuentes
$\dot{\mathbf{s}}, \dot{\mathbf{s}}(t)$	Vectores de derivadas primeras de las fuentes de interés
$\ddot{\mathbf{s}}, \ddot{\mathbf{s}}(t)$	Vectores de derivadas segundas de las fuentes de interés
\mathbf{v}	Autovectores de una matriz
$\mathbf{x}, \mathbf{x}(t)$	Vector de observaciones recibidas en los receptores
$\mathbf{y}, \mathbf{y}(t)$	Vector de estimaciones de las fuentes originales
\mathbf{A}	Matriz de mezclas instantáneas
$\mathbf{C}_{xx}, \mathbf{R}_{xx}(0)$	Matriz de autocovarianzas de las observaciones
$\mathbf{C}_{\tau}^x, \mathbf{R}_{xx}(\tau)$	Matriz espacial de autocovarianzas de las observaciones evaluada en τ
\mathbf{I}	Matriz identidad
$\mathbf{FT}[\cdot]$	Transformada de Fourier
\mathbf{G}	Matriz global de transferencia sin memoria
M	Número de receptores

N	Número de fuentes
N_{Fil}	Número de filtros utilizados en la versión avanzada del algoritmo
\mathbf{P}	Matriz de Permutación
$\mathbf{P}_{\mathbf{A}^\perp}$	Matriz de proyección sobre el subespacio ortogonal a las columnas de \mathbf{A}
T	Número de observaciones
\mathbf{R}_{xx}^h	Matriz de autocovarianzas de las observaciones filtradas con $h(t)$
\mathbb{R}^n	Espacio n-dimensional de los números reales
SNR	Relación señal a ruido recibida en los receptores
\mathbf{V}	Matriz de preblanqueo
\mathbf{W}	Matriz de separación instantánea

ÍNDICE

1	INTRODUCCIÓN	1
1.1	SEPARACIÓN Y EXTRACCIÓN CIEGA DE FUENTES.....	2
1.2	ALGORITMOS Y SOFTWARE.....	3
1.3	PLANTEAMIENTO Y ESTRUCTURA DEL PROYECTO.....	3
2	INTRODUCCIÓN A ICA Y A PCA	5
2.1	ANÁLISIS DE COMPONENTES PRINCIPALES.....	5
2.1.1	Introducción.....	5
2.1.2	Descripción del método.....	7
2.2	ANÁLISIS DE COMPONENTES INDEPENDIENTES.....	9
2.2.1	Introducción.....	9
2.2.2	Planteamiento del modelo en presencia de ruido.....	9
2.2.3	Ilustración del funcionamiento de los métodos ICA.....	11
2.2.4	Restricciones de los métodos ICA.....	13
2.2.5	Comparación entre ICA y PCA.....	15
2.3	RESUMEN DEL CAPÍTULO.....	16
3	SEPARACIÓN DE SEÑALES CON ESTRUCTURA TEMPORAL Y ESTUDIO DE LOS MODELOS PARA ICA	18
3.1	MÉTODOS DE SEPARACIÓN BASADOS EN LA ESTRUCTURA TEMPORAL.....	18

3.1.1	Introducción.....	18
3.1.2	Ilustración de las técnicas de separación mediante matrices de autocovarianzas.....	19
3.1.3	El algoritmo AMUSE.....	20
3.1.4	Limitaciones del algoritmo.....	22
3.2	ESTUDIO DE LOS DIFERENTES MODELOS DE SEÑAL EN ICA.....	22
3.2.1	Modelo de señal para mezclas instantáneas.....	22
3.2.2	Modelo de señal para mezclas convolutivas.....	23
3.2.3	Modelo de señal para pequeños retardos.....	25
3.3	RESUMEN DEL CAPÍTULO.....	26
4	DESCRIPCIÓN DEL ALGORITMO DE SEPARACIÓN DE MEZCLAS DE FUENTES CON RETARDOS	28
4.1	INTRODUCCIÓN.....	28
4.2	CONSIDERACIONES PREVIAS Y NOTACIÓN EMPLEADA.....	29
4.3	MODELADO MATEMÁTICO PARA N FUENTES Y M RECEPTORES.....	31
4.4	RESOLUCIÓN MATEMÁTICA DEL MODELO.....	34
4.5	ALGORITMO PARA LA OBTENCIÓN DE LA MATRIZ DE SEPARACIÓN.....	38
4.6	MEJORA DEL COMPORTAMIENTO FRENTE A LOS RETARDOS MEDIANTE UNA EXPANSIÓN EN SERIE DE TAYLOR DE SEGUNDO ORDEN.....	39
4.7	AMPLIACIONES Y MEJORAS DE LA VERSIÓN BÁSICA DEL ALGORITMO.....	42
4.7.1	Introducción.....	42
4.7.2	Descripción del método de extracción de fuentes usando varias matrices de autocovarianzas.....	42
4.7.3	El método de deflación.....	45
4.8	RESUMEN DEL CAPÍTULO.....	47

5	SIMULACIONES Y ESTUDIO DEL COMPORTAMIENTO DEL ALGORITMO	48
5.1	ILUSTRACIÓN DEL FUNCIONAMIENTO DE LA VERSIÓN BÁSICA DEL ALGORITMO.....	48
5.1.1	Introducción.....	48
5.1.2	Ejemplo de separación de mezclas de fuentes con retardos.....	48
5.1.3	Obtención de las derivadas de las fuentes originales.....	54
5.1.4	Comportamiento del algoritmo frente al ruido.....	58
5.1.4.1	Respuesta ante el ruido.....	58
5.1.4.2	Elección del filtro utilizado para la obtención de la segunda matriz de autocovarianzas.....	61
5.1.4.3	Comportamiento del algoritmo al variar el número de receptores.....	62
5.1.5	Comportamiento frente a los retardos de las fuentes.....	65
5.2	ESTUDIO DEL FUNCIONAMIENTO DE LA VERSIÓN AMPLIADA DEL ALGORITMO.....	70
5.2.1	Introducción.....	70
5.2.2	Ejemplo de separación de fuentes con retardos.....	70
5.2.3	Obtención de las derivadas de las fuentes originales.....	72
5.2.4	Comportamiento frente al ruido.....	73
5.2.4.1	Respuesta ante el ruido.....	73
5.2.4.2	Comportamiento del algoritmo al variar el número de receptores.....	77
5.2.5	Elección de los filtros utilizados y estudio del comportamiento del algoritmo al variar el número de filtros.....	78
5.2.6	Comportamiento frente a los retardos de las fuentes.....	80
5.3	COMPARACIÓN ENTRE LAS DOS VERSIONES DEL ALGORITMO.....	82
5.4	RESULTADOS OBTENIDOS AL UTILIZAR LA EXPANSIÓN EN SERIE DE TAYLOR DE SEGUNDO ORDEN.....	83
5.4.1	Introducción.....	83
5.4.2	Simulaciones realizadas y resultados obtenidos.....	84
5.4.3	Conclusiones sobre el método de separación propuesto.....	88

5.5	RESUMEN DEL CAPÍTULO Y CONCLUSIONES.....	89
6	CONCLUSIONES Y LÍNEAS FUTURAS	90
6.1	CONCLUSIONES.....	90
6.2	LÍNEAS FUTURAS DE INVESTIGACIÓN.....	91
A	DEMOSTRACIONES	92
A.1	DEPENDENCIAS ENTRE UN PROCESO ESTACIONARIO DE PRIMER ORDEN Y SU DERIVADA PRIMERA.....	92
A.2	LIMITACIONES DEL MÉTODO DE SEPARACIÓN PROPUESTO PARA EL CASO DE UN MEDIO QUE NO INTRODUCE ATENUACIÓN.....	94
B	RESOLUCIÓN DE PROBLEMAS ADICIONALES	96
B.1	CÁLCULO DE LOS AUTOVECTORES DE UNA MATRIZ.....	96
	BIBLIOGRAFÍA	98

CAPÍTULO 1

INTRODUCCIÓN

De manera natural, las personas somos capaces de seleccionar en un entorno ruidoso, en el que existen varias conversaciones, aquella que nos interesa. Se trata de un claro ejemplo de separación ciega de fuentes (BSS) conocido como *cocktail-party problem*. Entre las muchas aplicaciones de ICA, se encuentra la resolución del *cocktail-party problem* o separación de hablantes. El procedimiento de identificación de hablantes se lleva a cabo a partir de una serie de mezclas de fuentes recogidas en micrófonos dispersos e inmersos en un entorno ruidoso.

En el modelo básico de ICA se supone que las mezclas de las fuentes de interés, que se reciben en los receptores, se producen de forma instantánea y en ausencia de ruido. Un modelo más realista debería considerar la presencia de ruido y el hecho de que las mezclas no se producen de forma instantánea, porque las fuentes siempre sufrirán retardos en su propagación hacia los receptores. Por tanto, el modelo clásico de ICA, ya no será válido. Si queremos un modelo más próximo a la realidad, tendremos que recurrir al modelo de ICA para mezclas convolutivas, que tiene en cuenta la respuesta impulsiva del canal en el que se propagan las fuentes. Este modelo es mucho más exacto, pero tiene el inconveniente de que su resolución es mucho más compleja.

Veremos que si los retardos que sufren las señales en su propagación son pequeños, es posible tener en cuenta su efecto sin necesidad de utilizar el modelo de ICA convolutivo. Se demostrará que la resolución del problema de separación teniendo en cuenta los retardos puede llevarse a cabo con el modelo de mezclas instantáneas, lo cual facilitará la resolución del problema de forma considerable. Para ello, se considerarán las derivadas de las fuentes originales como nuevas fuentes que intervienen en el proceso de mezclas.

1.1 Separación y extracción ciega de fuentes

La separación ciega de fuentes (BSS) consiste en la identificación de un conjunto de N fuentes, de las que no se disponen datos a priori, a partir de un conjunto de mezclas de las mismas que se recogen en M sensores distintos. Ésta será la técnica utilizada en la versión básica del algoritmo propuesto.

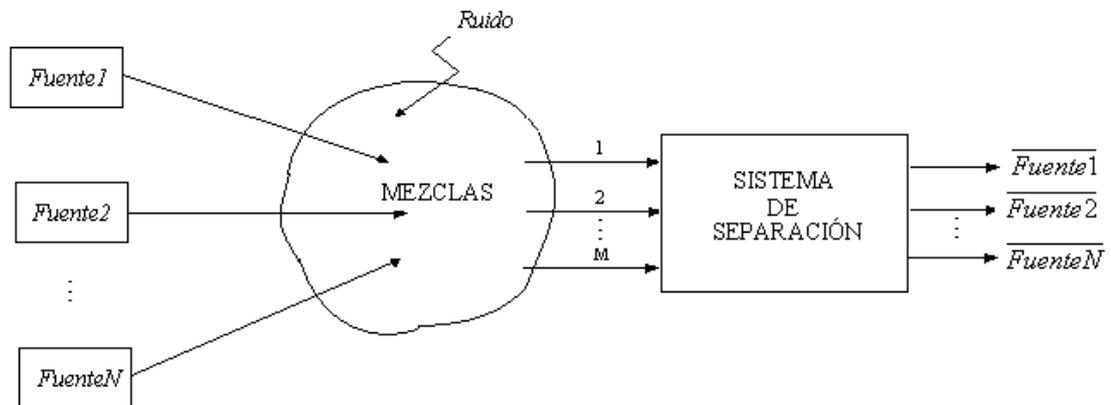


Figura 1.1: Esquema del proceso de separación ciega de fuentes

Las mezclas de fuentes que se recogen en los sensores pueden ser de diversos tipos. En función del tipo de mezcla recibida en los sensores, se utilizará un modelo u otro para la resolución del problema de separación ciega.

Una primera clasificación¹ de las mezclas es en mezclas lineales y no lineales. En las mezclas lineales, las observaciones son combinaciones lineales de las fuentes de interés. A su vez, las mezclas lineales pueden ser de dos tipos, instantáneas y convolutivas. En el presente proyecto, nos centraremos en las mezclas lineales e instantáneas. Se verá que, aunque el modelo instantáneo de mezclas no es adecuado para la identificación de fuentes cuando éstas son recibidas con retardos, bajo ciertas circunstancias esta aproximación puede ser válida. En los sistemas de mezclas lineales e instantáneas, la separación se realiza gracias a que las fuentes se combinan en cada sensor de forma diferente y aprovechando la independencia de las mismas. Esto es lo que se denomina diversidad espacial.

¹ Para una descripción más detallada, véase [4]

Por último, las mezclas pueden clasificarse también en mezclas con y sin ruido. En nuestro caso, se trabajará con mezclas con ruido, para modelar de forma más exacta el problema. El ruido con el que se reciben las fuentes se supondrá que es gaussiano e independiente de las fuentes.

La extracción ciega de fuentes (BSE) consiste en la separación ciega de sólo un conjunto de las N fuentes que producen las mezclas, a partir de dichas mezclas. La idea es extraer sólo las fuentes de interés que se encuentran en las mezclas recibidas. Una forma alternativa de obtener todas las fuentes de interés de una mezcla, es extraer las fuentes de la mezcla una a una y luego realizar la deflación sobre el vector de mezclas de la fuente extraída. En la versión ampliada del algoritmo propuesto para la separación de fuentes, se empleará esta técnica, ya que, tal como se verá, permite obtener ciertas ventajas sobre la BSS clásica.

1.2 Algoritmos y software

Para la separación de fuentes con pequeños retardos, se implementó un algoritmo en el lenguaje de programación MATLAB. Dicho algoritmo realiza la separación ciega de todas las fuentes recibidas en los sensores. Además se implementó una mejora del mismo, que extrae de las mezclas recibidas, una a una, las fuentes de interés. Cuando se extrae una fuente, se realiza la operación de deflación de la fuente extraída, eliminándola de las mezclas, para volver a extraer la siguiente fuente.

También se programó una función encargada de generar las mezclas de fuentes con retardos, con las que trabajan los algoritmos de separación implementados.

El código de las dos versiones del algoritmo implementadas y el de la función que genera las mezclas se adjuntan en la documentación anexa a este proyecto.

1.3 Planteamiento y estructura del proyecto

El objetivo del presente proyecto es la implementación y el estudio de un algoritmo que utiliza un modelo de mezclas lineales e instantáneas para la separación de fuentes, que se reciben con pequeños retardos. En las diversas simulaciones y estudios realizados, se considerará que se las mezclas recibidas son voces de distintos hablantes

(cocktail-party problem), aunque el algoritmo propuesto, puede ser extendido para el caso de otro tipo de señales que sean recibidas con distintos retardos.

El proyecto ha sido estructurado en 6 capítulos. En el capítulo 2, se presentan los fundamentos teóricos de PCA e ICA, necesarios para la comprensión de los algoritmos de separación que se explicarán en capítulos sucesivos.

En el capítulo 3, se analizará el caso de mezclas de señales con una estructura temporal, frente al caso de mezclas de variables aleatorias estadísticamente independientes. También se hará un análisis más detallado de los distintos modelos de mezclas lineales en ICA y se expondrá el modelo que se utilizará para modelar los retrasos de las fuentes.

En el capítulo 4, se explicará con detalle el algoritmo de separación propuesto. Así como de las hipótesis en las que se basa. También se explicarán las ampliaciones propuestas para la versión básica del algoritmo.

En el capítulo 5, se recogen y analizan todos los resultados obtenidos en las diversas simulaciones realizadas en distintas condiciones.

Finalmente, las conclusiones y posibles líneas futuras de investigación se recogen en el capítulo 6.

CAPÍTULO 2

INTRODUCCIÓN A ICA Y PCA

2.1 Análisis de componentes principales

2.1.1 Introducción

El Análisis de Componentes Principales (PCA), también llamado transformada de Karhunen-Loève, es una técnica estadística para el tratamiento de datos de forma compacta que elimina la redundancia de los datos originales. Se trata pues de obtener una representación de los datos lo mejor posible. Las técnicas PCA encuentran numerosas aplicaciones en diversas materias que requieren el tratamiento de datos (especialmente para la compresión). En nuestro caso, será una herramienta especialmente útil para el procesado previo de datos para la aplicación de técnicas de Análisis de Componentes Independientes (ICA).

Dado un vector aleatorio \mathbf{x} con n componentes, del cual conocemos una muestra $\mathbf{x}(1), \mathbf{x}(2), \dots, \mathbf{x}(T)$, el objetivo de PCA consiste en tratar de eliminar la redundancia existente entre cada una de sus componentes de forma que se obtenga un nuevo vector \mathbf{y} con m componentes, tal que $n \geq m$. Con esto conseguimos un vector de menor dimensión, sin redundancia entre sus componentes.

En PCA, no se tiene en cuenta el tipo de función de densidad que poseen los vectores. Sólo supondremos que se pueden estimar los estadísticos de primer y segundo orden de la muestra.

PCA trata de buscar un nuevo sistema de ejes coordenados formado por una base ortogonal para representar al vector \mathbf{x} de forma que se maximiza la varianza de las

proyecciones del vector en dicho sistema. Así conseguimos que las componentes de \mathbf{x} en el nuevo sistema no estén correladas.

Para ilustrar el funcionamiento de PCA, en la siguiente figura se representa la distribución de probabilidad conjunta de dos variables aleatorias x_1 y x_2 obtenidas cada una de ellas a partir de la combinación lineal de otras dos variables aleatorias s_1 y s_2 independientes entre sí.

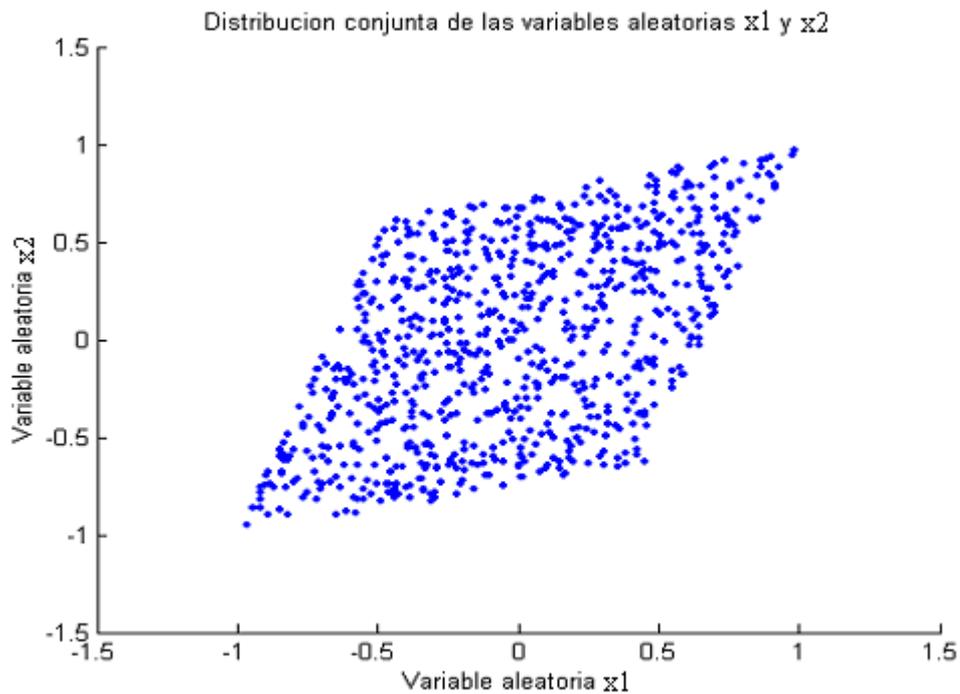


Figura 2.1: Distribución de probabilidad conjunta de la variables aleatorias x_1 y x_2 .

Si a las variables aleatorias x_1 y x_2 se les realiza un procesamiento PCA, las nuevas variables que se obtienen no estarán correladas entre sí y tendrán una distribución de probabilidad conjunta como se ilustra a continuación:

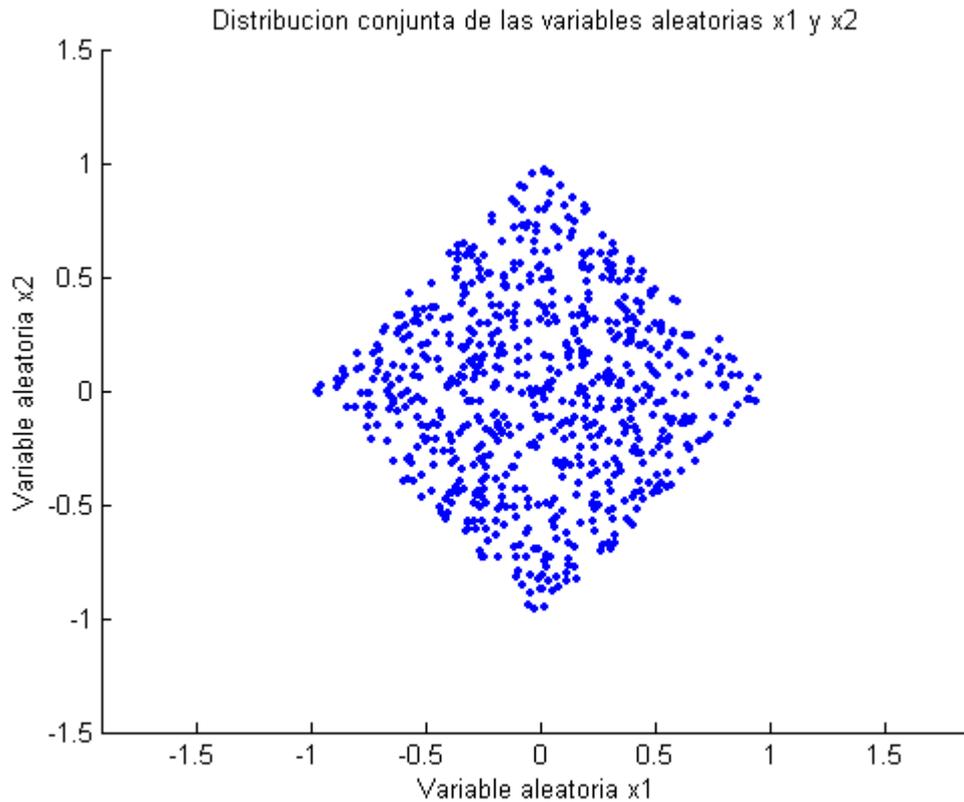


Figura 2.2: Nueva distribución de probabilidad conjunta de las variables tras un procesamiento PCA.

Tal como puede verse en la *figura 2.2* las nuevas variables obtenidas estarán no correladas entre sí y su función de distribución conjunta estará contenida en un cuadrado.

Las técnicas PCA persiguen objetivos distintos que las técnicas ICA. Sin embargo, se verá que PCA puede ser útil como un procesamiento previo al uso de técnicas ICA. También pueden usarse para blanquear o decorrelar los datos antes de aplicarles un determinado procesado.

2.1.2 Descripción del método

En PCA se trata de buscar las componentes principales de las observaciones que consisten en los autovalores de la matriz de covarianzas de éstas. Para poder aplicar el método, debemos centrar el vector \mathbf{x} . Para ello, simplemente, tenemos que sustraerle la media:

$$\mathbf{x} = \mathbf{x} - E\{\mathbf{x}\} \quad (2.1)$$

Denotaremos como \mathbf{y} la nueva representación de los datos. Se trata de un vector de dimensión m . En general, se cumplirá que $m \leq n$ siendo n la dimensión de los datos originales.

La nueva representación de los datos tras aplicar las técnicas PCA vendrá dada por:

$$\mathbf{y} = \mathbf{w}^T \mathbf{x} \quad (2.2)$$

Donde \mathbf{w} es una matriz cuyas filas son vectores de dimensión n , tales que maximizan la varianza entre las componentes del vector \mathbf{y} . Para dicha matriz sus columnas serán ortogonales entre sí, es decir:

$$\mathbf{w}_i^T \mathbf{w}_j = \delta_{ij} \quad (2.3)$$

Puede demostrarse, véase [8], que las columnas de la matriz \mathbf{w} coinciden con una base ortogonal expandida a partir de los autovectores $\mathbf{e}_1, \mathbf{e}_2, \dots, \mathbf{e}_m$ de la matriz de covarianzas, \mathbf{C}_x , del vector \mathbf{x} . Dichos autovectores serán distintos entre sí y se ordenarán en función del valor de sus autovalores $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_m$. La mayor parte de la información estará contenida en el primer autovector \mathbf{e}_1 . La cantidad de información contenida en cada autovalor será mayor cuanto mayor sea su autovalor asociado, de forma que en las primeras componentes se concentrará la mayor parte de la información. Al valor $y_1 = \mathbf{e}_1^T \mathbf{x}$ se le denomina *componente principal* de \mathbf{x} , y es la que contiene más información de todas.

En el caso de que $m < n$, estaremos cometiendo un error en la representación de los datos. Como ventaja estaremos comprimiendo la información contenida en el vector inicial. El error cuadrático mínimo que podemos cometer se dará cuando tomemos una base ortogonal expandida a partir de los m autovectores $\mathbf{e}_1, \mathbf{e}_2, \dots, \mathbf{e}_m$ de la matriz de covarianzas. Dicho error vendrá dado por:

$$J_{MSE}^{PCA} = \sum_{i=m+1}^n d_i \quad (2.4)$$

Si $m = n$ no se cometerá error alguno en la representación de los datos en la nueva base formada por los autovectores de la matriz de covarianzas de los datos. Este caso será de especial interés, ya que al aplicar un análisis PCA a los datos, conseguiremos decorrelarlos.

2.2 Análisis de componentes independientes

2.2.1 Introducción

El Análisis de Componentes Independientes (ICA) es una técnica pensada para obtener o extraer una representación de cada una de las distintas componentes o señales independientes entre sí que forman parte de una señal de mezcla. Veremos que esto será posible siempre que las distintas señales que componen la mezcla sean independientes entre sí; es decir, el valor de una de ellas no está relacionado con el de las otras, y que no más de una de ellas tenga una distribución de tipo gaussiana.

Entre las numerosas aplicaciones de ICA, nos centraremos en las aplicaciones para BSS (Separación ciega de fuentes) que consisten en la recuperación de un conjunto de señales a partir de mezclas lineales de ellas sin saber nada sobre las fuentes que queremos recuperar.

2.2.2 Planteamiento del modelo en presencia de ruido

Consideraremos que observamos m variables aleatorias en presencia de ruido a las que denotaremos como x_1, x_2, \dots, x_m . Estas variables aleatorias son el resultado de combinaciones de otras variables aleatorias, a las que denotaremos como s_1, s_2, \dots, s_n , que queremos obtener a partir de las variables que hemos observado.

En general, podremos expresar cada una de las variables aleatorias que hemos observado de la forma:

$$x_i = a_{i1}s_1 + a_{i2}s_2 + \dots + a_{in}s_n + n_i \quad \text{con } i = 1, \dots, m \quad (2.5)$$

Donde los coeficientes a_{ij} son coeficientes reales *a priori* desconocidos, con $i = 1, \dots, m$ y $j = 1, \dots, n$, y n_i es el ruido recibido en el i -ésimo receptor. En general, supondremos que las variables aleatorias s_i son estadísticamente independientes y que, como máximo, una de ellas tiene una distribución de tipo gaussiana. También supondremos que el número de variables aleatorias observadas cumple que $m > n$ en el caso de que las observaciones se hagan en presencia de ruido y que $m \geq n$ en ausencia de ruido.

En el presente documento se empleará notación vectorial para referirse a las fuentes y a las observaciones. Denotaremos como \mathbf{x} al vector que contiene cada una de las mezclas observadas x_1, x_2, \dots, x_m , como \mathbf{s} al vector que contiene cada una de las componentes independientes s_1, s_2, \dots, s_n que queremos separar y como \mathbf{n} al vector que contiene las fuentes de ruido recibidas en cada sensor. Asimismo denotaremos como \mathbf{A} la matriz de mezclas cuyos elementos son los coeficientes a_{ij} . De forma compacta, podemos expresar el proceso de mezcla de la siguiente forma:

$$\mathbf{x} = \mathbf{A}\mathbf{s} + \mathbf{n} \quad (2.6)$$

En la siguiente figura se ilustra gráficamente el proceso de mezcla y de estimación de las señales originales a partir de sus mezclas:

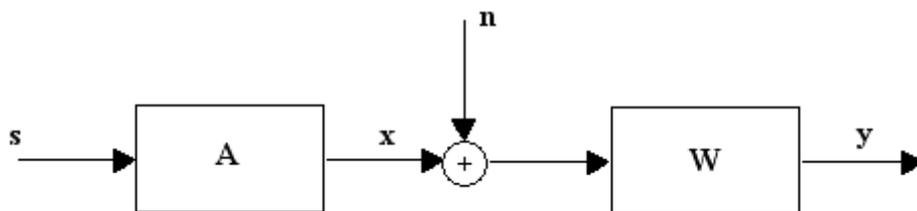


Figura 2.3: Esquema del proceso de mezclas y de separación

Nuestro objetivo será determinar las componentes independientes (Ics) a partir de las observaciones. Para ello, trataremos de buscar una estimación de inversa de la matriz de mezclas \mathbf{W} , que nos permita recuperar las señales originales a partir de las observaciones. Normalmente la matriz \mathbf{A} será invertible, aunque si el número de observaciones es mayor que el de fuentes (tendremos redundancia) la matriz de mezclas no será cuadrada. En estos casos, la identificación de las fuentes a partir de las mezclas también es posible y nos limitaremos a buscar una matriz que al multiplicarla por la matriz de mezclas nos dé la matriz identidad.

2.2.3 Ilustración del funcionamiento de los métodos ICA

Para mostrar de forma cualitativa el correcto funcionamiento de ICA, consideraremos dos fuentes estadísticamente independientes entre sí, s_1 y s_2 , distribuidas de forma uniforme en un cuadrado.

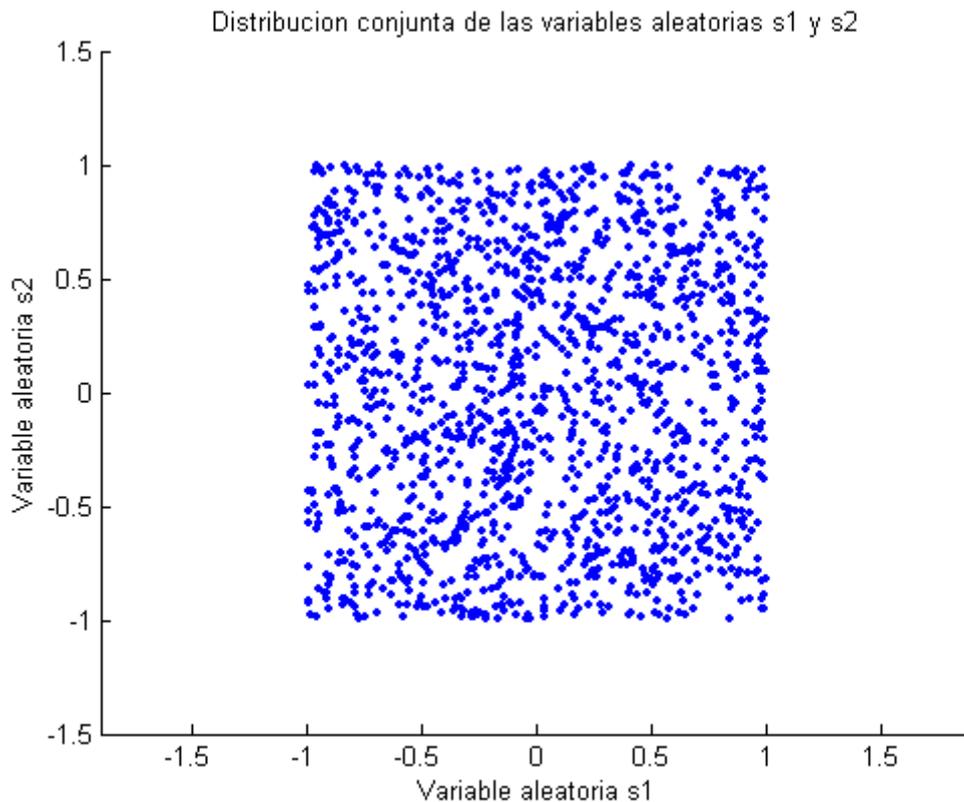


Figura 2.4: Distribución conjunta de las variables aleatorias s_1 y s_2 . Cada una de ellas con distribución uniforme. Eje horizontal s_1 . Eje vertical s_2

Podemos ver como las dos variables aleatorias son independientes entre sí. Si fijamos el valor de una de ellas, no podemos determinar el valor de la otra.

Ahora supongamos que mezclamos ambas variables aleatorias mediante una matriz de mezcla \mathbf{A} de dimensión 2×2 para obtener otras dos nuevas variables aleatorias x_1 y x_2 . Tal como puede verse en la distribución conjunta de ambas variables, las dos nuevas variables aleatorias no son independientes entre sí. Si fijamos el valor de una de ellas, el valor de la otra estará condicionado por el valor que tome la primera.

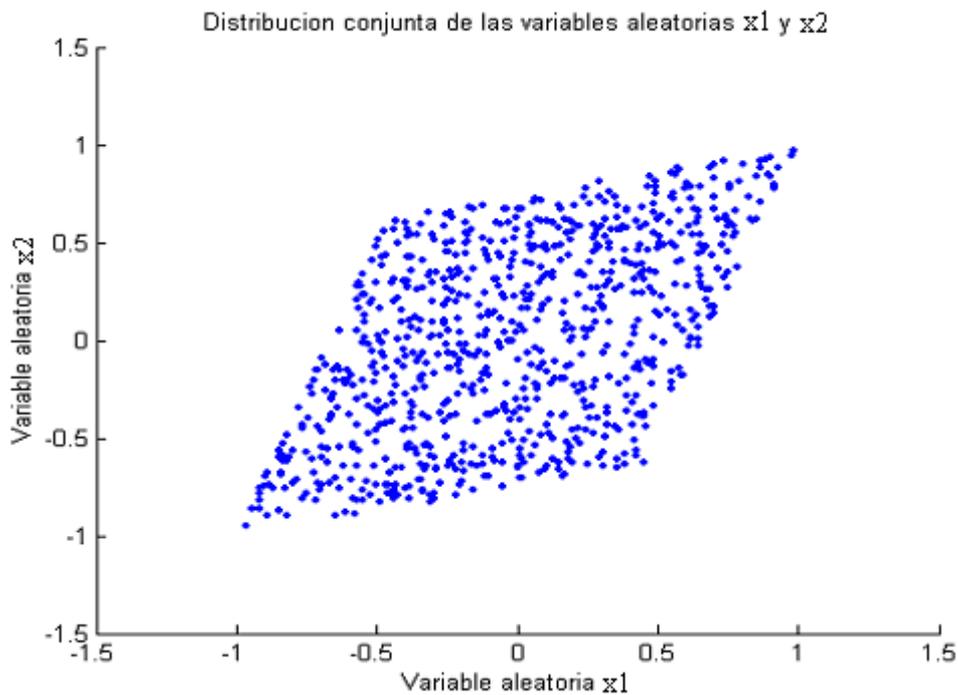


Figura 2.5: Distribución conjunta de las variables x_1 y x_2 . Eje horizontal: variable x_1 . Eje vertical: variable x_2

El problema de obtención de las componentes independientes originales debe hacerse a partir de las observaciones y consiste en estimar la matriz \mathbf{A} a partir de las observaciones. De forma intuitiva podemos ver que las direcciones del paralelepípedo de la *Figura 2.5* coinciden con las columnas de la matriz \mathbf{A} . Este método no es útil para separar las fuentes, sin embargo podemos ver de forma cualitativa que el problema de separación tiene solución si las variables de mezcla son estadísticamente independientes. En apartados posteriores, se comentarán en detalle algunos métodos analíticos para la obtención de las distintas ICs de una mezcla utilizados en el presente documento.

2.2.4 Restricciones de los métodos ICA

Existen varias limitaciones que poseen los métodos ICA y que es conveniente destacar. Estas limitaciones o condicionantes no son demasiado restrictivos y son asumibles en la mayoría de las aplicaciones de interés. A continuación se describen las principales restricciones de ICA:

- **Las componentes independientes deben ser estadísticamente independientes**

Como ya se vio en apartados anteriores, necesitamos que las variables aleatorias que se mezclan s_1, s_2, \dots, s_n sean independientes entre sí. Dos variables aleatorias son independientes entre sí cuando una de ellas no da información de la otra. Matemáticamente, dado un conjunto de variables aleatorias, se dice que son estadísticamente independientes si su función de densidad conjunta es igual al producto de las funciones de densidad marginales:

$$p(s_1, s_2, \dots, s_n) = p_1(s_1)p_2(s_2)\dots p_n(s_n) \quad (2.7)$$

Este condicionante, en la práctica, no resulta ser demasiado restrictivo: en la mayoría de las aplicaciones, las señales que sufren el proceso de mezcla son independientes. Así, en el caso que posteriormente se tratará de señales de voz de distintas personas, esta suposición se cumple.

- **Las componentes independientes deben tener distribuciones no gaussianas**

Para que la identificación de las distintas componentes sea posible, es necesario que tengan funciones de distribución no gaussianas. Con funciones de distribución gaussianas, la identificación de la matriz \mathbf{A} no es siempre posible. Para establecer las condiciones de separación cuando existen fuentes gaussianas, partiremos del siguiente teorema:

Teorema 2.1: *Teorema de Darmois-Skitovich:* Sean dos combinaciones lineales de fuentes estadísticamente independientes:

$$y_1[n] = G_{11}s_1[n] + G_{12}s_2[n]$$

$$y_2[n] = G_{21}s_1[n] + G_{22}s_2[n]$$

Si $y_1[n]$ e $y_2[n]$ son independientes y $\{G_{1j} \neq 0, G_{2j} \neq 0\}$, entonces s_j es gaussiana.

Para variables gaussianas, el hecho de que no estén correladas implica independencia. Por tanto, la propiedad de independencia en la que se basa ICA no puede ser utilizada para la identificación de variables gaussianas. Así, en el caso de que la matriz \mathbf{A} sea ortogonal, bien por casualidad o porque hayamos preblanqueado los datos, la función de distribución conjunta de las fuentes originales y de las mezclas son idénticas, por lo que no puede obtenerse ninguna información adicional de las fuentes a partir de las mezclas y no podrá determinarse la matriz \mathbf{A} .

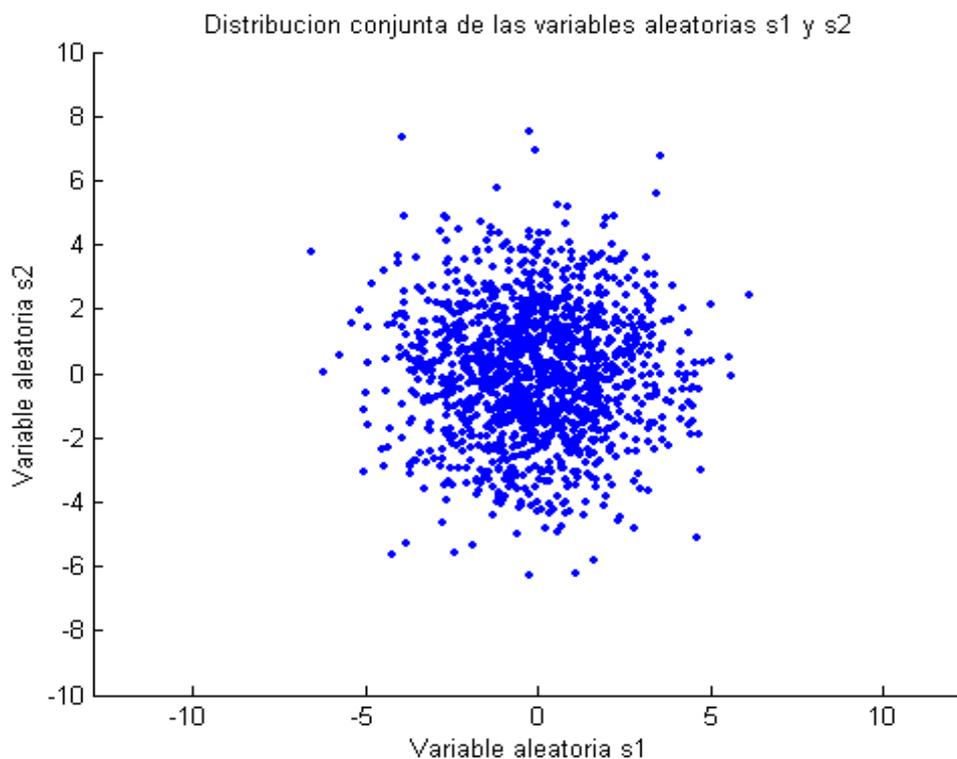


Figura 2.6: Representación de la función de distribución de dos variables aleatorias de tipo gaussiano. Dicha distribución coincide con la de las mezclas decorreladas de las fuentes gaussianas.

- **No es posible determinar las amplitudes de las componentes independientes ni el orden de las mismas.**

Las amplitudes no pueden determinarse porque las matrices \mathbf{A} y \mathbf{s} son desconocidas. Un factor que multiplique al vector de fuentes \mathbf{s} puede ser cancelado, dividiendo una de las columnas de \mathbf{A} por el mismo factor. Por tanto, ICA podrá determinar las Ics originales salvo una constante multiplicativa. Decir que dicha constante puede tener signo negativo e invertir las señales. Esto no supone una limitación para la mayoría de los casos de interés.

Tampoco puede determinarse el orden de las Ics. La razón es la misma de antes: desconocemos la matriz de mezclas y las fuentes originales. Podemos escribir las señales recibidas en los receptores de la forma $\mathbf{x} = \mathbf{A}\mathbf{P}^{-1}\mathbf{P}\mathbf{s}$, donde \mathbf{P} es una matriz de permutación. Entonces vemos que el valor de la nueva matriz de mezclas vendrá dado por $\mathbf{A}\mathbf{P}^{-1}$. Es decir, podremos determinar la matriz de mezclas salvo una permutación. Esta limitación no supondrá un problema, ya que normalmente nos centraremos en obtener todas las componentes independientes. Decir que mediante técnicas como la deflación (véase implementación del algoritmo de separación ciega con pequeños retardos) es posible extraer las señales originales ordenadas de mayor a menor energía.

2.2.5 Comparación entre ICA y PCA

Aunque ambos métodos están relacionados entre sí, persiguen objetivos distintos: el objetivo de PCA es decorrelar los datos y reducir el número de variables, mientras el de ICA es obtener por separado cada una de las componentes independientes que forman parte de una mezcla.

Por otra parte ICA usa el concepto de independencia que es más restrictivo que el de correlación, ya que si dos variables son independientes serán no correladas, pero no necesariamente al revés.

Dos variables aleatorias son no correladas si su covarianza es cero:

$$\text{cov}(y_1, y_2) = E\{y_1 y_2\} - E\{y_1\} E\{y_2\} = 0 \quad (2.8)$$

Y si dos variables aleatorias son independientes porque cumplen la condición (2.7), también serán no correladas porque en ese caso:

$$E(y_1 y_2) = E(y_1) E(y_2) \quad (2.9)$$

Aunque ambas técnicas persiguen objetivos distintos y se basan en propiedades distintas, no son excluyentes. Las técnicas PCA pueden usarse para preblanquear los datos antes de aplicarles un análisis ICA y simplificar así de forma considerable el problema. El preblanqueo de los datos consiste en multiplicar los datos recibidos por una matriz \mathbf{V} de forma que los nuevos datos recibidos no estén correlados entre sí. Los nuevos datos obtenidos tras preblanquear vendrán dados por:

$$\mathbf{z} = \mathbf{V}\mathbf{x} = \mathbf{V}\mathbf{A}\mathbf{s} \quad (2.10)$$

Y se cumplirá que $E\{\mathbf{z}\mathbf{z}^T\} = \mathbf{I}$, porque al preblanquear los nuevos datos que se obtienen ya no estarán correlados.

Con el preblanqueo de los datos se consigue que la matriz de mezclas, $\mathbf{V}\mathbf{A}$, sea ortogonal haciendo más sencilla su identificación, ya que sólo tenemos que buscar en el espacio de matrices ortogonales. Al ortogonalizar la matriz \mathbf{A} , se pasa de tener n^2 grados de libertad a tener $\frac{n(n-1)}{2}$ grados de libertad, por lo que se simplifica extraordinariamente el problema de identificación.

2.3 Resumen del capítulo

En el presente capítulo, se han analizado dos técnicas para el tratamiento de datos: PCA e ICA. Ambas técnicas persiguen objetivos distintos y están basadas en propiedades distintas. La primera de ellas está basada busca eliminar la redundancia de

un conjunto de datos intentando eliminar la correlación entre las distintas muestras que conforman los datos. La segunda de ellas está basada en la propiedad de independencia y busca la obtención de una serie de fuentes estadísticamente independientes entre sí a partir de una mezcla de ellas.

Pese a las diferencias que presentan ICA y PCA, no son técnicas excluyentes. En capítulos posteriores, se utilizarán las técnicas PCA para preblanquear los datos y facilitar la resolución del modelo ICA.

CAPÍTULO 3

SEPARACIÓN DE SEÑALES CON

ESTRUCTURA TEMPORAL Y ESTUDIO

DE LOS MODELOS PARA ICA

3.1 Métodos de separación basados en la estructura temporal

3.1.1 Introducción

Hasta ahora, en el modelo básico de ICA planteado se consideraba una mezcla lineal de variables aleatorias estadísticamente independientes. En este modelo básico, el orden de las muestras del vector \mathbf{x} no tiene especial relevancia. Se puede alterar el orden de las muestras del vector y el modelo seguirá siendo igualmente válido.

En el presente proyecto, a partir de ahora, nos centraremos en el caso en que tenemos una mezcla de señales con dependencia temporal, $s_i(t)$ con $t = 1, \dots, T$. Las señales recibidas pueden expresarse matricialmente de la forma:

$$\mathbf{x}(t) = \mathbf{A}\mathbf{s}(t) \quad (3.1)$$

En este caso, el orden de las muestras tiene especial relevancia. Tendremos una estructura temporal que nos permitirá obtener información adicional para la resolución del problema de separación. Las componentes independientes son señales con dependencias temporales. Así la matriz de autocovarianzas de una señal temporal será

distinta si se evalúa para distintos retardos de la señal. Con lo cual, podremos obtener varias matrices de autocovarianzas a partir de los datos y conseguir a partir de ellas información adicional para la resolución del problema. De esta forma se evitará el uso de estadísticos de orden superior.

Para poder hacer uso de esta información adicional, deben darse dos condiciones:

- Las componentes independientes deben tener distintas autocovarianzas. Además las autocovarianzas deben ser distintas de cero.
- Las varianzas de las componentes independientes son no estacionarias.

En la inmensa mayoría de los casos de interés, estas suposiciones se cumplen y no resultan ser demasiado restrictivas.

3.1.2 Ilustración de las técnicas de separación mediante matrices de autocovarianzas

En el modelo básico de ICA sólo podemos separar las señales temporales originales a partir de las señales de mezcla usando la matriz de autocovarianzas en el caso de que los datos recibidos no estén correlados, es decir, $E\{x_i(t)x_j(t)^T\} = 0$ si $i \neq j$. En ese caso particular, tendremos que:

$$C_{xx} = E\{\mathbf{x}(t)\mathbf{x}(t)^T\} = E\{A\mathbf{s}\mathbf{s}^T A^T\} = AC_{ss}A^T \quad (3.2)$$

Donde la matriz C_{xx} es diagonal porque las fuentes son estadísticamente independientes. Las columnas de la matriz \mathbf{A} serán iguales a los autovectores de la matriz C_{xx} .

En el caso de los datos recibidos estén correlados, no podemos determinar la matriz \mathbf{A} de la forma anterior. Podríamos multiplicar los datos recibidos por una matriz de preblanqueo \mathbf{V} y decorrelarlos, sin embargo, existen infinitas matrices de preblanqueo que nos permiten obtener los datos, por lo que el método propuesto sólo

nos permitiría determinar la matriz \mathbf{VA} (matriz de mezclas). Esto es lo que ocurre cuando se intenta separar variables gaussianas. Para variables gaussianas, decorrelación implica independencia. Las fuentes recibidas nunca serán independientes puesto que son una combinación lineal de las señales originales que queremos separar, por tanto, las señales recibidas serán correladas y por la razón antes expuesta, sólo se podrá determinar la matriz \mathbf{VA} , pero no la matriz \mathbf{A} . Esto no supone una limitación a la hora de obtener las fuentes originales a partir de sus mezclas puesto que para ello sólo se necesita conocer la matriz de mezclas \mathbf{VA} .

En estos casos, tendrían que usarse estadísticos de orden superior para determinar la matriz \mathbf{A} . Una alternativa al uso de estadísticos de orden superior es utilizar otra matriz de covarianzas en un instante de tiempo distinto al cero y así poder determinar la matriz de separación, ya que se cumplirá que:

$$C_{\tau}^x = E\{\mathbf{x}(t)\mathbf{x}(t-\tau)\} = E\{A\mathbf{s}\mathbf{s}^T A^T\} = AC_{\tau}^s A^T \quad (3.3)$$

C_{τ}^s es una matriz diagonal, debido a que se cumple que $E\{x_i(t)x_j(t-\tau)\} = 0$ para todo $i \neq j$ porque el valor de la señal recibida en el i -ésimo receptor en un instante t no depende de la señal recibida en el receptor j -ésimo en un instante $t-\tau$ distinto al anterior. Así con el uso de matrices de autocovarianzas evaluadas en distintos instantes de tiempo, se obtiene información adicional que nos permite no tener que recurrir al uso de estadísticos de mayor orden.

3.1.3 El algoritmo AMUSE

El algoritmo AMUSE [14] es uno de los métodos de separación ciega más significativos entre aquellos que obtienen la matriz de separación haciendo uso de matrices de autocovarianzas. Además guarda una profunda similitud con otros algoritmos de separación de fuentes, como el que posteriormente se describirá para la separación de fuentes con pequeños retrasos, de ahí, que se describa con detalle en el presente capítulo.

El algoritmo utiliza un solo retardo temporal, τ , para el cálculo de las matrices de autocovarianzas. Usualmente, por simplicidad, se hace $\tau = 1$.

Sean $\mathbf{z}(t) = \mathbf{V}\mathbf{x}(t)$ las fuentes recibidas y preblanqueadas. Para ellas se cumplirá:

$$\mathbf{W}\mathbf{z}(t) = \mathbf{s}(t) \quad (3.4)$$

$$\mathbf{W}\mathbf{z}(t-\tau) = \mathbf{s}(t-\tau) \quad (3.5)$$

Donde $\mathbf{W}^{-1} = \mathbf{V}\mathbf{A}$ y \mathbf{V} es la matriz de preblanqueo. Consideremos la matriz

$$\bar{\mathbf{C}}_{\tau}^z = \frac{1}{2} \left[\mathbf{C}_{\tau}^z + (\mathbf{C}_{\tau}^z)^T \right] \quad (3.6)$$

Donde $\mathbf{C}_{\tau}^z = \mathbf{W}^T E \left\{ \mathbf{s}(t) \mathbf{s}(t-\tau)^T \right\} \mathbf{W} = \mathbf{W}^T \mathbf{D}\mathbf{W}$ es la matriz de covarianzas en el instante τ (generalmente se hace $\tau = 1$). Esta matriz es simétrica puesto que la matriz \mathbf{D} es diagonal, pues se asume que las fuentes a separar son estadísticamente independientes. Por tanto, podemos escribir:

$$\mathbf{C}_{\tau}^z = \frac{1}{2} \mathbf{W}^T \left[E \left\{ \mathbf{s}(t) \mathbf{s}(t-\tau)^T \right\} + E \left\{ \mathbf{s}(t-\tau)^T \mathbf{s}(t) \right\} \right] \mathbf{W} = \mathbf{W}^T \bar{\mathbf{C}}_{\tau}^s \mathbf{W} \quad (3.7)$$

La matriz $\bar{\mathbf{C}}_{\tau}^s$ es diagonal puesto que las fuentes son independientes. De (3.7) podemos deducir que las filas de la matriz \mathbf{W} vienen dadas por los autovectores de la matriz $\bar{\mathbf{C}}_{\tau}^z$.

En resumen, para el cálculo de la matriz \mathbf{W} se siguen los siguientes pasos:

- Los datos recibidos $\mathbf{x}(t)$ se preblanquean y se obtiene $\mathbf{z}(t) = \mathbf{V}\mathbf{x}(t)$
- A partir de $\mathbf{z}(t)$, se obtiene $\mathbf{C}_{\tau}^z = E \left\{ \mathbf{z}(t) \mathbf{z}(t-\tau)^T \right\}$, normalmente para $\tau = 1$, y posteriormente $\bar{\mathbf{C}}_{\tau}^z = \frac{1}{2} \left[\mathbf{C}_{\tau}^z + (\mathbf{C}_{\tau}^z)^T \right]$

- Se calculan los autovectores de la matriz $\overline{\mathbf{C}}_\tau^z$. Las filas de la matriz \mathbf{W} serán iguales a los autovectores.
- Finalmente se obtienen las fuentes sabiendo que $\mathbf{s}(t) = \mathbf{Wz}(t)$

3.1.4 Limitaciones del algoritmo

El algoritmo sólo funciona bien en el caso de que la matriz $\overline{\mathbf{C}}_\tau^z$ no tenga autovectores repetidos. En este caso, no podrán determinarse las fuentes correspondientes a autovalores repetidos. Escogiendo un retardo τ adecuado podemos resolver el problema si las señales $s_i(t)$ tienen distintas autocovarianzas (densidad espectral).

3.2 Estudio de los diferentes modelos de señal en ICA

3.2.1 Modelo de señal para mezclas instantáneas

Se trata del modelo básico considerado hasta ahora. Es el caso en el que las fuentes en su propagación desde el transmisor hasta el receptor sólo sufren una mezcla lineal entre ellas y la multiplicación por un factor constante. En dicho modelo denotamos como $\mathbf{s}(t) = (s_1(t), s_2(t), \dots, s_N(t))^T$ con $t = 1, \dots, T$ al vector de fuentes independientes entre sí y de media cero. \mathbf{A} es una matriz de dimensión $M \times N$ que permite modelar la combinación lineal de las señales cuando se transmite por el canal. En el receptor tenemos M observaciones que denotamos mediante el vector $\mathbf{x}(t) = (x_1(t), x_2(t), \dots, x_N(t))^T$ con $t = 1, \dots, T$.

Matricialmente podemos expresar el vector de observaciones de la forma:

$$\mathbf{x}(t) = \mathbf{A}\mathbf{s}(t) \quad (3.8)$$

En la siguiente figura aparece representado el modelo básico:

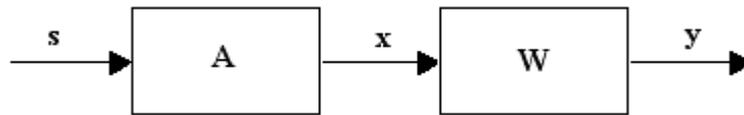


Figura 3.1: Modelo para el proceso de mezcla y separación

La matriz \mathbf{W} es la matriz de separación. Al multiplicar dicha matriz por la de mezclas, debe obtenerse el producto de una matriz diagonal por una matriz permutación, es decir, $\mathbf{WA} = \mathbf{DP}$. Ello se debe a las limitaciones de ICA, comentadas en el capítulo 2, para determinar el orden de las componentes independientes y su amplitud. Por último, el vector \mathbf{y} es la estimación de las fuentes originales que queremos separar.

A este modelo, al igual que los otros dos que se expondrán a continuación, se les puede añadir ruido. Por simplicidad, no se ha hecho, ya que en el estudio de los diferentes modelos de señal para ICA, nos centraremos en el tipo de respuesta impulsiva del canal.

3.2.2 Modelo de señal para mezclas convolutivas

En este caso las señales se propagan por un canal con memoria caracterizado por una respuesta impulsiva, que se supone desconocida. Las señales recibidas ya no sufren una simple combinación entre ellas como en el caso anterior. Este modelo es especialmente útil en los casos donde la propagación entre las fuentes y los sensores se ve afectada por la existencia de multitrayectos. Se trata de un modelo matemáticamente más complejo, ya que es necesario manejar secuencias de matrices tanto en el sistema de mezclas como en el de separación. El modelado matemático es el siguiente:

Sean $s_i(t)$ con $i=1,2,\dots,N$, $t=1,\dots,T$ los procesos generadores de la secuencia de fuentes $\mathbf{s}(t) = (s_1(t), s_2(t), \dots, s_N(t))^T$. Se supone que se cumplen las siguientes hipótesis:

- Los procesos $s_i(t)$ son estacionarios en sentido amplio (WSS) y mutuamente independientes entre sí.
- Cada uno de los procesos $s_i(t)$ está constituido de variables aleatorias temporalmente independientes e idénticamente distribuidas.
- Se supone que el sistema de mezcla es MIMO y lineal e invariante en el tiempo (LTI). Por tanto, está caracterizado por una respuesta al impulso matricial $\mathbf{A}(t)$. El esquema del proceso de mezclas es el mismo que el de la *figura 3.1*, pero ahora las observaciones son el resultado de la convolución de la secuencia de fuentes con la respuesta al impulso del sistema de mezcla, es decir:

$$\mathbf{x}(t) = \mathbf{A}(t) * \mathbf{s}(t) = \sum_{k=-\infty}^{\infty} \mathbf{A}(k) \mathbf{s}(t-k) \quad (3.9)$$

Se supone que el sistema de separación también es LTI y, por tanto, queda determinado por la respuesta al impulso matricial $\mathbf{W}(t)$ de longitud $T = 2l_w + 1$. Así el vector de salidas se construye como:

$$\mathbf{y}(t) = \mathbf{W}(t) * \mathbf{x}(t) = \sum_{k=-l_w}^{l_w} \mathbf{W}(k) \mathbf{x}(t-k) \quad (3.10)$$

La respuesta al impulso del sistema conjunto de mezcla y separación viene dada por

$$\mathbf{G}(t) = \mathbf{W}(t) * \mathbf{A}(t) \quad (3.11)$$

Es trivial comprobar que el modelo de mezcla instantánea es un caso particular del convolutivo en el que los sistemas de mezclas y de separación no tienen memoria:

$$\mathbf{A}(t) = \mathbf{A}(0)\delta(t) \text{ y } \mathbf{W}(t) = \mathbf{W}(0)\delta(t) \quad (3.12)$$

3.2.3 Modelo de señal para pequeños retardos

Este será el modelo utilizado en los algoritmos que posteriormente se expondrán con detalle. Se trata de un caso intermedio entre los dos anteriores. En dicho modelo se supone que en el receptor se recibe una mezcla de las señales idénticas a las transmitidas, con la salvedad de un retardo y una atenuación debido a su propagación por el medio. Podemos considerar este modelo como una extensión del modelo básico ICA en el cual se consideran, además, los retardos debido a la propagación de las señales transmitidas.

La respuesta impulsiva del medio de propagación desde la fuente j hasta el receptor i es de la forma:

$$h_{ij}(t) = a_{ij}\delta(t - \tau_{ij}) \text{ con } i = 1, \dots, M \quad j = 1, \dots, N \text{ y } |a_{ij}| < 1 \quad (3.13)$$

Al igual que en el modelo convolutivo, el sistema estará caracterizado por una respuesta al impulso matricial $\mathbf{A}(t)$. Las observaciones pueden expresarse matemáticamente de la forma:

$$\mathbf{x}(t) = \mathbf{A}(t) * \mathbf{s}(t) = \sum_{k=-\infty}^{\infty} \mathbf{A}(k)\mathbf{s}(t-k) \quad (3.14)$$

$$\text{con } \mathbf{A}(t) = \begin{bmatrix} a_{11}\delta(t - \tau_{11}) & \cdots & a_{1N}\delta(t - \tau_{1N}) \\ \vdots & & \vdots \\ a_{M1}\delta(t - \tau_{M1}) & \cdots & a_{MN}\delta(t - \tau_{MN}) \end{bmatrix} \quad (3.15)$$

El sistema de separación podrá expresarse de la misma forma que en el modelo anterior. Supondremos que es LTI y, por tanto, queda determinado por una respuesta al

impulso matricial $\mathbf{W}(t)$ de dimensiones $N \times M$ y de longitud $T = 2l_w + 1$. El valor de las salidas se obtiene como:

$$\mathbf{y}(t) = \mathbf{W}(t) * \mathbf{x}(t) = \sum_{k=-l_w}^{l_w} \mathbf{W}(k) \mathbf{x}(t-k) \quad (3.16)$$

La respuesta al impulso del sistema conjunto de mezcla y separación viene dada por

$$\mathbf{G}(t) = \mathbf{W}(t) * \mathbf{A}(t) \quad (3.17)$$

En el algoritmo que se implementó para la separación de fuentes con retardos, tal como se describe en este modelo, pero la resolución del mismo se hace utilizando el modelo de ICA instantáneo. Para ello, se verá en el capítulo siguiente que, bajo ciertas condiciones, es posible la resolución del modelo con pequeños retardos considerando que las mezclas se producen de forma instantánea, y a la vez, considerando el efecto de los retardos.

3.3 Resumen del capítulo

En el modelo básico de ICA se considera una mezcla lineal de variables aleatorias estadísticamente independientes. En este capítulo, se ha extendido el modelo básico al caso de una mezcla de señales temporales. Se tendrá entonces una estructura temporal que permitirá obtener información adicional para la identificación de las fuentes a partir de las mezclas. Así las matrices de autocovarianzas serán distintas si las evaluamos para distintos retardos de la señales recibidas.

El uso de estas matrices de autocovarianzas permitirá idear algoritmos que eviten el uso de estadísticos de orden superior, consiguiéndose así mayor simplicidad en la resolución del modelo. De entre todos estos algoritmos, se ha destacado el algoritmo AMUSE, debido al gran parecido que presentan con él los algoritmos que usan matrices de autocovarianzas evaluadas en distintos instantes temporales. En el capítulo posterior, se verá la profunda similitud que presenta con el algoritmo AMUSE el algoritmo propuesto para la separación de fuentes con pequeños retardos.

Finamente se han presentado los diversos modelos de mezclas que existen en ICA y se ha introducido el modelo con retardos, que modela el proceso de mezclas para el cual se ha desarrollado el algoritmo de separación que se describirá en el capítulo siguiente.

CAPÍTULO 4

DESCRIPCIÓN DEL ALGORITMO DE SEPARACIÓN DE MEZCLAS DE FUENTES CON RETARDOS²

4.1 Introducción

Se desea proceder a la separación de N fuentes, que pueden estar situadas en distintos puntos. Estas fuentes son grabadas simultáneamente en M receptores, con $M \geq N$, separados una pequeña distancia entre sí. Se trata de un problema de separación de fuentes de audio (cocktail-party problem), con una salvedad: En cada sensor las fuentes son recibidas con distintos retardos. Es una aplicación de interés, porque al tratar de separar fuentes de audio, éstas serán recibidas en los receptores con pequeños retardos, bien porque los sensores suelen colocarse en distintas posiciones, bien porque las fuentes pueden estar separadas o por los dos motivos anteriores. Los retardos serán distintos porque las fuentes recorrerán trayectos distintos al propagarse desde las fuentes a los receptores. Consideraremos que la velocidad de propagación de las fuentes en el medio es constante y que la respuesta impulsiva del medio es, a priori, desconocida.

² Para una descripción más detallada, véase [1].

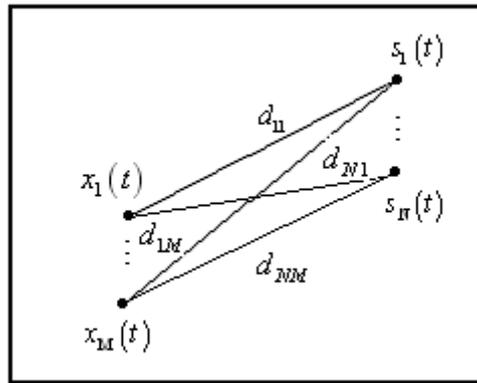


Figura 4.1: Ilustración del problema de separación con M receptores y N fuentes

Debido a los retardos con los que son recibidas las fuentes, el modelo de ICA lineal sin memoria ya no será válido. El modelo a utilizar será el expuesto en el capítulo anterior para fuentes con pequeños retardos.

Se verá cómo la separación de dichas fuentes es posible en el caso de que los retardos de las fuentes sean pequeños (posteriormente se describirá qué entendemos por retraso pequeño).

Para proceder a la resolución del problema, se considerarán las derivadas de primer orden de las fuentes originales como fuentes adicionales y se usarán estadísticos de segundo orden. El problema se resolverá como uno de mezclas instantáneas, en el que se consideran los efectos de los retardos, al introducirse las derivadas. De esta forma, evitamos el uso del modelo ICA para mezclas convolutivas, de mucha mayor complejidad matemática.

4.2 Consideraciones previas y notación empleada

Consideraremos que las fuentes de audio que queremos separar $s_j(t)$ con $j = 1, \dots, N$ son estadísticamente independientes y poseen diferente densidad espectral de potencia. Además dichas fuentes son limitadas en banda, lo cual siempre ocurre cuando se trabaja con señales físicas reales, y son emitidas desde un punto determinado. Las observaciones en los receptores se denotarán como $x_i(t)$ con $i = 1, \dots, M$.

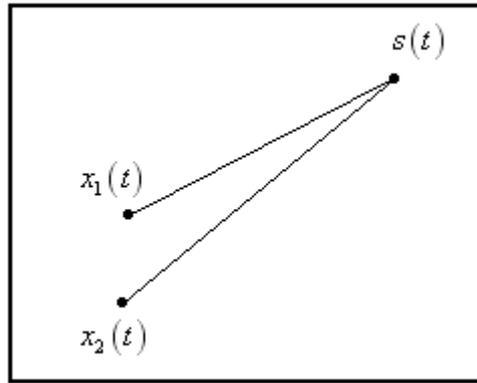


Figura 4.2: Ilustración del problema de separación con dos receptores y una fuente

Para ilustrar el problema, consideraremos que sólo se emite **una fuente puntual** $s(t)$, tal como se ilustra en la *figura 4.2* (luego se estudiará el caso en el que se emitan N fuentes de forma simultánea). Las señales recibidas en los receptores pueden expresarse de la forma:

$$x_i(t) = h_i(t) * s(t) \quad (4.1)$$

Donde $h_i(t) = \alpha_i \delta(t - \tau_i)$ con $|\alpha| < 1$, es decir, estamos considerando un medio en el que sólo se introduce un retardo y una atenuación en la propagación de la señal. Las señales recibidas pueden expresarse de la forma:

$$\begin{aligned} x_1(t) &= \alpha'_1 s(t - \tau'_1) \\ &\vdots \\ x_M(t) &= \alpha'_M s(t - \tau'_M) \end{aligned} \quad (4.2)$$

Puesto que se supone desconocida la respuesta impulsiva del medio, podemos expresar las señales recibidas con una atenuación y un retardo relativo con respecto a la señal recibida en uno de los sensores, que consideramos de referencia (se tomará el primero de ellos como referencia). Por tanto, las señales recibidas se expresarán de la forma:

$$\begin{aligned}
 x_1(t) &= s(t) \\
 x_2(t) &= \alpha_2 s(t - \tau_2) \\
 &\vdots \\
 x_M(t) &= \alpha_M s(t - \tau_M)
 \end{aligned}
 \tag{4.3}$$

Para la resolución del problema de separación se supone que los receptores en los que se graban las señales recibidas y las fuentes están próximos entre sí, de forma que los retrasos que se obtienen son pequeños. La señal $s(t - \tau)$ puede expresarse en serie de potencias de la forma:

$$s(t - \tau) = s(t) - \tau \dot{s}(t) + R_2(\tau) \tag{4.4}$$

Y como los retardos son pequeños, puede hacerse la aproximación:

$$s(t - \tau) \approx s(t) - \tau \dot{s}(t) \tag{4.5}$$

Por tanto, podemos expresar las observaciones en los sensores distintos al de referencia de la forma:

$$x_i(t) \approx \alpha_i s(t) - \alpha_i \tau_i \dot{s}(t) \text{ con } i \neq 1 \tag{4.6}$$

4.3 Modelado matemático para N fuentes y M receptores

Esta situación básica considerada hasta ahora puede generalizarse para el caso de N fuentes emitidas de forma simultánea y situadas en posiciones distintas (véase la *figura 4.1*). Se trata de un problema de separación de una mezcla de fuentes con retrasos. Su resolución será igual a la de un problema clásico de separación ciega de mezclas instantáneas en el que las fuentes a identificar son las de interés, $\mathbf{s}(t)$, y sus derivadas, $\dot{\mathbf{s}}(t)$.

El vector de fuentes recibidas $\mathbf{x}(t) = (x_1(t), \dots, x_M(t))^T$ puede expresarse de la forma:

$$\mathbf{x}(t) = \mathbf{A} \begin{bmatrix} \mathbf{s}(t) \\ \dot{\mathbf{s}}(t) \end{bmatrix} + \mathbf{n}(t) = \mathbf{A}\tilde{\mathbf{s}}(t) + \mathbf{n}(t) \quad \text{con} \quad \mathbf{s}(t) = (s_1(t), \dots, s_N(t))^T \quad (4.7)$$

Donde $\mathbf{s}(t) = (s_1(t), \dots, s_N(t))^T$ es el vector que contiene las fuentes estadísticamente independientes que queremos separar y $\mathbf{n}(t) = (n_1(t), \dots, n_M(t))^T$ es el vector de ruido y contiene las fuentes de ruido recibidas en cada uno de los receptores. Dicho ruido será independiente de las fuentes y blanco.

Por último, \mathbf{A} es la matriz de mezclas sin memoria de dimensión $M \times 2N$ con $M \geq N$.

El problema de separación consistirá en la obtención de una matriz de separación \mathbf{W} , tal que $\mathbf{WA} = \mathbf{DP}$ donde \mathbf{D} es una matriz diagonal e invertible y \mathbf{P} es una matriz de permutación.

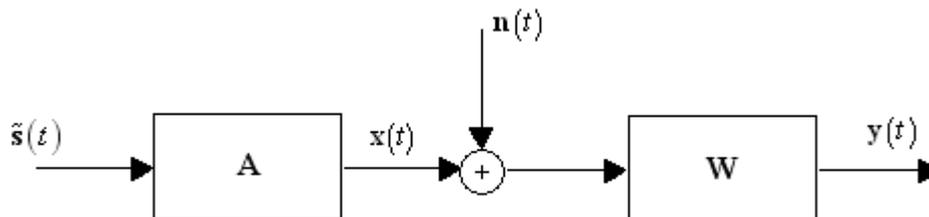


Figura 4.2: Modelo del proceso de mezclas y de separación.

Si consideramos M sensores $x_1(t), \dots, x_M(t)$ y N fuentes $s_1(t), \dots, s_M(t)$, de forma que la j -ésima fuente es recibida en el i -ésimo receptor con un retardo debido a su propagación τ_{ij} y una atenuación c_{ij} , las señales recibidas en los sensores pueden expresarse de la forma, con respecto al de referencia:

$$x_1(t) = \sum_{j=1}^N s_j(t) + n_1(t) \quad \text{para el sensor de referencia con } i = 1 \quad (4.8)$$

$$x_i(t) = \sum_{j=1}^N c_{ij} s_j(t - \tau_{ij}) + n_i(t) \text{ para el resto de sensores con } i = 2, \dots, M \quad (4.9)$$

La transformada de Fourier del término $s_j(t - \tau_{ij})$ es:

$$FT[s_j(t - \tau_{ij})] = s_j(f) e^{-j2\pi f \tau_{ij}} \text{ donde } f \text{ es la frecuencia} \quad (4.10)$$

La exponencial puede expresarse en serie de potencias de la forma:

$$e^{-j2\pi f \tau_{ij}} = \sum_{k=0}^{\infty} \frac{(-j2\pi f \tau_{ij})^k}{k!} \quad (4.11)$$

Se considerarán retrasos pequeños cuando los términos del desarrollo en serie de orden mayor que uno sean despreciables frente al primero, es decir:

$$\frac{(2\pi f \tau_{ij})^2}{2} \ll 1 \quad (4.12)$$

Si denotamos como f_{\max} la máxima frecuencia en la que las fuentes son distintas de cero, $s_j(f) \neq 0$, la condición de retardo pequeño podemos expresarla de la forma:

$$\tau_{ij}^2 \ll \frac{1}{2\pi^2 f_{\max}^2} \quad \forall i, j \quad (4.13)$$

En un contexto real, los sensores tendrán un ancho de banda determinado $[f_1, f_2]$ y que deberá cumplirse que $f_1 < f_{\max} < f_2$.

Bajo estas condiciones, la observación i -ésima puede ser aproximada de la forma:

$$\begin{aligned} x_i(t) \approx & c_{i,1} s_1(t) - c_{i,1} \tau_{i,1} \frac{ds_1(t)}{dt} + c_{i,2} s_2(t) - c_{i,2} \tau_{i,2} \frac{ds_2(t)}{dt} + \dots \\ & \dots + c_{i,N} s_N(t) - c_{i,N} \tau_{i,N} \frac{ds_N(t)}{dt} + n_i(t) \end{aligned} \quad (4.14)$$

Y la expresión matricial $\mathbf{x}(t) = \mathbf{A}\tilde{\mathbf{s}}(t) + \mathbf{n}(t)$ puede ser escrita:

$$\mathbf{x}(t) \approx \mathbf{A}_1 \mathbf{s}(t) + \mathbf{A}_2 \dot{\mathbf{s}}(t) + \mathbf{n}(t) \quad (4.15)$$

Donde

$$\mathbf{A}_1 = \begin{bmatrix} 1 & \cdots & 1 & \cdots & 1 \\ c_{2,1} & \cdots & c_{2,j} & \cdots & c_{2,N} \\ \vdots & & \vdots & & \vdots \\ c_{M,1} & \cdots & c_{M,j} & \cdots & c_{M,N} \end{bmatrix} \quad \mathbf{A}_2 = - \begin{bmatrix} 0 & \cdots & 0 & \cdots & 0 \\ c_{2,1}\tau_{2,1} & \cdots & c_{2,j}\tau_{2,j} & \cdots & c_{2,N}\tau_{2,N} \\ \vdots & & \vdots & & \vdots \\ c_{M,1}\tau_{M,1} & \cdots & c_{M,j}\tau_{M,j} & \cdots & c_{M,N}\tau_{M,N} \end{bmatrix}$$

Otra forma de expresar el vector de observaciones será:

$$\mathbf{x}(t) \approx [\mathbf{A}_1 \quad \mathbf{A}_2] \begin{bmatrix} \mathbf{s}(t) \\ \dot{\mathbf{s}}(t) \end{bmatrix} + \mathbf{n}(t) = \mathbf{A} \begin{bmatrix} \mathbf{s}(t) \\ \dot{\mathbf{s}}(t) \end{bmatrix} + \mathbf{n}(t) \quad (4.16)$$

Siendo $\mathbf{A} = [\mathbf{A}_1 \quad \mathbf{A}_2]$ la matriz de mezclas instantánea de dimensiones $M \times 2N$. En general, debe cumplirse $M > 2N$ y que $M \geq 2N$ Si no existe ruido. El objetivo será la identificación de la matriz \mathbf{A} . Nótese que, a diferencia del modelado básico de ICA propuesto en el capítulo 2, la matriz de mezclas no será invertible, ya que si $M > 2N$ la matriz no será cuadrada. En estos casos, se tendrá cierta redundancia que facilitará la identificación de las mezclas.

4.4 Resolución matemática del modelo

El método empleado para obtener la matriz de separación es similar al del algoritmo AMUSE [14] y está basado en el uso de dos matrices de autocovarianzas distintas, obtenidas a partir de las observaciones.

La matriz de autocovarianzas para un retardo τ de las señales recibidas es:

$$\mathbf{R}_{\mathbf{xx}}(\tau) = E[\mathbf{x}(t)\mathbf{x}(t+\tau)^T] \quad (4.17)$$

Y según (4.16) podemos expresar la matriz anterior de la siguiente forma:

$$\mathbf{R}_{xx}(\tau) \approx \mathbf{A}E \left[\begin{bmatrix} \mathbf{s}(t) \\ \dot{\mathbf{s}}(t) \end{bmatrix} \begin{bmatrix} \mathbf{s}(t+\tau) \\ \dot{\mathbf{s}}(t+\tau) \end{bmatrix}^T \right] \mathbf{A}^T + E \left[\mathbf{n}(t)\mathbf{n}(t+\tau)^T \right] \approx \mathbf{A} \begin{bmatrix} \mathbf{R}_{ss}(\tau) & \mathbf{R}_{s\dot{s}}(\tau) \\ \mathbf{R}_{\dot{s}s}(\tau) & \mathbf{R}_{\dot{s}\dot{s}}(\tau) \end{bmatrix}^T \mathbf{A}^T + \mathbf{R}_{nn}(\tau)$$

Para un retraso nulo, $\tau = 0$, las fuentes $s_i(t)$ y sus derivadas $\dot{s}_i(t)$ no están correladas, por tanto, las matrices $\mathbf{R}_{ss}(0)$ y $\mathbf{R}_{\dot{s}\dot{s}}(0)$ son nulas³ y las matrices $\mathbf{R}_{ss}(0)$ y $\mathbf{R}_{\dot{s}\dot{s}}(0)$ son diagonales. La expresión anterior puede escribirse de la forma:

$$\mathbf{R}_{xx}(0) = \mathbf{A}\mathbf{D}_0\mathbf{A}^T + \mathbf{R}_{nn}(0) \quad \text{Siendo } \mathbf{D}_0 \text{ y } \mathbf{R}_{nn}(0) \text{ matrices diagonales}$$

Si filtramos las señales recibidas con un filtro de respuesta impulsiva $h(t)$, la matriz de autocovarianzas de las observaciones en el instante τ podemos expresarla como:

$$\mathbf{R}_{xx}^h(\tau) \approx \mathbf{A} \begin{bmatrix} \mathbf{R}_{ss}^h(\tau) & \mathbf{R}_{s\dot{s}}^h(\tau) \\ \mathbf{R}_{\dot{s}s}^h(\tau) & \mathbf{R}_{\dot{s}\dot{s}}^h(\tau) \end{bmatrix} \mathbf{A}^T + \mathbf{R}_{nn}^h(\tau) \quad (4.18)$$

Al igual que antes, las matrices $\mathbf{R}_{ss}^h(0)$ y $\mathbf{R}_{\dot{s}\dot{s}}^h(0)$ son nulas y las matrices $\mathbf{R}_{ss}^h(0)$ y $\mathbf{R}_{\dot{s}\dot{s}}^h(0)$ son diagonales. Por tanto:

$$\mathbf{R}_{xx}^h(0) = \mathbf{A}\mathbf{D}_0^h\mathbf{A}^T + \mathbf{R}_{nn}^h(0) \quad \text{Siendo } \mathbf{D}_0^h \text{ una matriz diagonal.}$$

En cuanto al ruido, supondremos que es blanco y que tiene la misma varianza σ^2 en cada sensor. Bajo estas suposiciones, los términos de ruido de las ecuaciones anteriores quedan:

$$\mathbf{R}_{nn}(\tau) = \sigma^2 \delta(\tau) \mathbf{I}_M \quad (4.19)$$

³ Véase demostración en el apéndice A

$$\mathbf{R}_{nn}^h(0) = \sigma^2 h(t) * \delta(t) \Big|_{t=0} = \sigma^2 h(0) \mathbf{I}_M \quad (4.20)$$

Para la resolución del modelo, se escogerá un filtro con respuesta impulsiva nula en cero, $h(0)=0$, y así cancelar los términos de ruido en la segunda matriz de autocovarianzas. Las matrices de autocovarianzas quedarán de la forma:

$$\mathbf{R}_{xx}(0) = \mathbf{A} \mathbf{D}_0 \mathbf{A}^T + \sigma^2 \mathbf{I}_M \quad (4.21)$$

$$\mathbf{R}_{xx}^h(0) = \mathbf{A} \mathbf{D}_0^h \mathbf{A}^T \quad (4.22)$$

Si descomponemos en valores singulares la matriz $\mathbf{R}_{xx}(0)$ obtenemos:

$$\mathbf{R}_{xx}(0) = [\mathbf{u}_1, \dots, \mathbf{u}_M] \mathbf{L} [\mathbf{u}_1, \dots, \mathbf{u}_M]^T \quad (4.23)$$

\mathbf{L} será una matriz diagonal de dimensión $M \times M$ de la forma $\mathbf{L} = \text{diag}(\lambda_1^2 + \sigma^2, \dots, \lambda_{2N}^2 + \sigma^2, \sigma^2, \dots, \sigma^2)$. Los últimos $M - 2N$ términos son iguales a la varianza del ruido σ^2 . Por lo que tomando su media puede hacerse una estimación de σ^2 . Tal como puede verse en la expresión de la matriz \mathbf{L} , en el caso de que haya ruido, necesitamos que $M > 2N$ para poder determinar la varianza del mismo.

Una vez estimada σ^2 , la parte de $\mathbf{R}_{xx}(0)$ debida a las fuentes puede expresarse:

$$[\mathbf{u}_1, \dots, \mathbf{u}_M] [\mathbf{L} - \sigma^2 \mathbf{I}_M] [\mathbf{u}_1, \dots, \mathbf{u}_M]^T = \mathbf{A} \mathbf{D}_0 \mathbf{A}^T \quad (4.24)$$

Siendo $\mathbf{L} - \sigma^2 \mathbf{I}_M = \text{diag}(\lambda_1^2, \dots, \lambda_{2N}^2, 0, \dots, 0)$.

Para la resolución del problema, se usarán dos matrices auxiliares: La matriz $\mathbf{U} = [\mathbf{u}_1, \dots, \mathbf{u}_{2N}]$ de dimensión $M \times 2N$ y la matriz $\mathbf{\Lambda} = \text{diag}(\lambda_1^2, \dots, \lambda_{2N}^2)$. De acuerdo con la ecuación (4.24), tenemos:

$$\mathbf{\Lambda} = \mathbf{U}^T \mathbf{A} \mathbf{D}_0 \mathbf{A}^T \mathbf{U} \quad (4.25)$$

Como $\mathbf{\Lambda}$ no tiene términos nulos en la diagonal, la matriz $\mathbf{U}^T \mathbf{M}$ de dimensión $2N \times 2N$ es invertible y puede escribirse:

$$\mathbf{\Lambda}^{-1} = [\mathbf{A}^T \mathbf{U}]^{-1} \mathbf{D}_0^{-1} [\mathbf{U}^T \mathbf{A}]^{-1} \quad (4.26)$$

Seguidamente definimos la matriz $\mathbf{\Gamma} = \mathbf{U}^T \mathbf{R}_{xx}^h(0) \mathbf{U} = \mathbf{U}^T \mathbf{A} \mathbf{D}_0^h \mathbf{A}^T \mathbf{U}$ y la matriz $\mathbf{R} = \mathbf{\Lambda}^{-1} \mathbf{\Gamma}$. Desarrollando obtenemos:

$$\mathbf{R} = [\mathbf{A}^T \mathbf{U}]^{-1} \mathbf{D}_0^{-1} [\mathbf{U}^T \mathbf{A}]^{-1} \mathbf{\Gamma} \quad (4.27)$$

Y desarrollando la expresión anterior, llegamos a que:

$$\mathbf{R} = [\mathbf{A}^T \mathbf{U}]^{-1} \mathbf{D}_0^{-1} [\mathbf{U}^T \mathbf{A}]^{-1} \mathbf{U}^T \mathbf{A} \mathbf{D}_0^h \mathbf{A}^T \mathbf{U} = [\mathbf{A}^T \mathbf{U}]^{-1} \mathbf{D}_0^{-1} \mathbf{D}_0^h \mathbf{A}^T \mathbf{U} \quad (4.28)$$

De aquí se deduce que $[\mathbf{A}^T \mathbf{U}]^{-1}$ es la matriz de autovectores de \mathbf{R} . El filtro $h(t)$ debe escogerse de forma que los autovectores de la matriz \mathbf{R} , sean todos distintos.

Si llamamos $\mathbf{V} = [\mathbf{A}^T \mathbf{R}]^{-1}$. La matriz de separación buscada es⁴

$$\mathbf{W} = \mathbf{V}^T \mathbf{U}^T \quad (4.29)$$

⁴ La demostración es sencilla: $\mathbf{W} \mathbf{A} = \left[[\mathbf{A}^T \mathbf{U}]^{-1} \right]^T \mathbf{U}^T \mathbf{A} = [\mathbf{U}^T \mathbf{A}]^{-1} [\mathbf{U}^T \mathbf{A}] = \mathbf{I}_{2N}$

4.5 Algoritmo para la obtención de la matriz de separación

El algoritmo encargado de obtener la matriz de separación y las fuentes originales a partir de las mezclas deberá realizar los siguientes pasos:

- 1) A partir de las observaciones de los sensores, $\mathbf{x}(t)$, se obtiene la matriz de autocovarianzas $\mathbf{R}_{\mathbf{xx}}(0)$.
- 2) Se realiza la descomposición en valores singulares de la matriz anterior ordenando los autovalores de mayor a menor.

$$\mathbf{R}_{\mathbf{xx}}(0) = [\mathbf{u}_1, \dots, \mathbf{u}_M] \mathbf{L} [\mathbf{u}_1, \dots, \mathbf{u}_M]^T \quad (4.30)$$

- 3) Se estima la varianza del ruido σ^2 . Su valor será igual a la media de los últimos $M - 2N$ valores singulares de $\mathbf{R}_{\mathbf{xx}}(0)$.
- 4) A partir de la matriz $\mathbf{L} = \text{diag}(\lambda_1^2 + \sigma^2, \dots, \lambda_{2N}^2 + \sigma^2, \sigma^2, \dots, \sigma^2)$ se obtiene $\mathbf{\Lambda} = \text{diag}(\lambda_1^2, \dots, \lambda_{2N}^2)$ y se forma la matriz $\mathbf{U} = [\mathbf{u}_1, \dots, \mathbf{u}_{2N}]$.
- 5) Se escoge un filtro con $h(t)$ impar, con $h(0) = 0$ y se obtiene la matriz de autocovarianzas de las observaciones filtrada.

$$\mathbf{R}_{\mathbf{xx}}^h(0) = h(t) * \mathbf{R}_{\mathbf{xx}}(0) \quad (4.31)$$

- 6) Computamos la matriz $\mathbf{R} = \mathbf{\Lambda}^{-1} \mathbf{U}^T \mathbf{R}_{\mathbf{xx}}^h(0) \mathbf{U}$
- 7) Calculamos \mathbf{V} , la matriz de autovectores de \mathbf{R} . La matriz de separación será $\mathbf{W} = \mathbf{V}^T \mathbf{U}^T$

8) Finalmente se obtienen las estimaciones de las fuentes que queremos separar:

$$\mathbf{y}(t) = \mathbf{W}\mathbf{x}(t) \quad (4.32)$$

4.6 Mejora del comportamiento frente a los retardos mediante una expansión en serie de Taylor de segundo orden

En el caso particular de que todas las fuentes en su propagación hacia los receptores sufran la misma atenuación⁵, es decir, los coeficientes c_{ij} de la ecuación (4.14) sean todos iguales a la unidad, puede utilizarse un método alternativo, basado en el anterior, para la estimación de las derivadas de las fuentes que han sido transmitidas. Dicho método nos va a permitir introducir retardos mayores que los impuestos por la ecuación (4.13) y consiste en operar con las observaciones antes de utilizar el algoritmo de separación propuesto.

Puesto que desconocemos la respuesta impulsiva del medio, podemos expresar los retardos con los que se reciben las fuentes en los receptores con respecto a uno de ellos, que consideraremos de referencia. Por defecto, consideraremos que este receptor es el primero y que es el que está más próximo a las fuentes, por lo que todos los retardos relativos, del resto de receptores respecto a él, serán positivos. Las fuentes recibidas en el receptores distintos al de referencia, $i = 2, \dots, M$, pueden expresarse matemáticamente de la siguiente forma:

$$x_1(t) = \sum_{j=1}^N s_j(t - \tau_{i,j}) + n_1(t) \quad (4.33)$$

Y si expandimos en serie de potencias los sumandos de la expresión anterior, despreciando los términos mayores de segundo orden, se obtendrá la siguiente aproximación para las observaciones en los receptores distintos al de referencia:

⁵ Esto puede ocurrir si la distancias entre fuentes y receptores son similares.

$$\begin{aligned}
 x_i(t) \approx & s_1(t) - \tau_{i,1} \frac{ds_1(t)}{dt} + \frac{\tau_{i,1}^2}{2} \frac{d^2s_1(t)}{dt^2} + s_2(t) - \tau_{i,2} \frac{ds_2(t)}{dt} + \frac{\tau_{i,2}^2}{2} \frac{d^2s_2(t)}{dt^2} + \dots \\
 & \dots + s_N(t) - \tau_{i,N} \frac{ds_N(t)}{dt} + \frac{\tau_{i,N}^2}{2} \frac{d^2s_N(t)}{dt^2} + n_i(t)
 \end{aligned} \quad (4.34)$$

Y la observación en el receptor de referencia, $i=1$, podrá expresarse de la siguiente forma:

$$x_1(t) = s_1(t) + s_2(t) + \dots + s_N(t) + n_1(t) \quad (4.35)$$

Si a las observaciones en receptores distintos al de referencia, $i=2, \dots, M$, le restamos las observaciones recibidas en el receptor de referencia, las observaciones relativas $\forall i = [2, \dots, M]$ pueden expresarse como sigue:

$$\begin{aligned}
 \tilde{x}_i(t) \approx & -\tau_{i,1} \frac{ds_1(t)}{dt} + \frac{\tau_{i,1}^2}{2} \frac{d^2s_1(t)}{dt^2} - \tau_{i,2} \frac{ds_2(t)}{dt} + \frac{\tau_{i,2}^2}{2} \frac{d^2s_2(t)}{dt^2} + \dots \\
 & \dots - \tau_{i,N} \frac{ds_N(t)}{dt} + \frac{\tau_{i,N}^2}{2} \frac{d^2s_N(t)}{dt^2} + n_i(t)
 \end{aligned} \quad (4.36)$$

Matricialmente las observaciones en los receptores distintos al de referencia pueden expresarse así:

$$\tilde{\mathbf{x}}(t) \approx [\mathbf{A}_3 \quad \mathbf{A}_4] \begin{bmatrix} \dot{\mathbf{x}}(t) \\ \ddot{\mathbf{x}}(t) \end{bmatrix} + \mathbf{n}(t) \quad (4.37)$$

La matrices \mathbf{A}_3 y \mathbf{A}_4 son de dimensiones $(M-1) \times N$ y de la forma:

$$\mathbf{A}_3 = - \begin{bmatrix} \tau_{2,1} & \cdots & \tau_{2,j} & \cdots & \tau_{2,N} \\ \vdots & & \vdots & & \vdots \\ \tau_{M,1} & \cdots & \tau_{M,j} & \cdots & \tau_{M,N} \end{bmatrix} \quad (4.38)$$

$$\mathbf{A}_4 = \frac{1}{2} \begin{bmatrix} \tau_{2,1}^2 & \cdots & \tau_{2,j}^2 & \cdots & \tau_{2,N}^2 \\ \vdots & & \vdots & & \vdots \\ \tau_{M,1}^2 & \cdots & \tau_{M,j}^2 & \cdots & \tau_{M,N}^2 \end{bmatrix} \quad (4.39)$$

La matriz $[\mathbf{A}_3 \ \mathbf{A}_4]$, de dimensión $(M-1) \times 2N$, es la matriz de mezclas instantáneas de las derivadas primera y segunda de las fuentes.

Al haberse introducido términos de segundo orden en el desarrollo en serie de Taylor, la condición de pequeños retardos puede expresarse ahora de la siguiente forma:

$$\tau_{i,j}^2 \ll \frac{3}{2\pi^2 V_M^2} \quad \forall i, j \quad (4.40)$$

Comparándola con la condición de pequeños retardos para el caso general, vemos que se pueden introducir retardos mayores, por lo que el método propuesto puede ser una alternativa al anterior cuando se verifiquen las condiciones comentadas anteriormente y se deseen introducir retardos mayores que no cumplan la condición descrita en (4.13).

La separación de las fuentes a partir de las observaciones obtenidas en los receptores distintos al de referencia, puede realizarse utilizando el método anterior. El algoritmo devolverá la estimación de las derivadas primeras y no la de las fuentes originales. Esto no supone una limitación en muchas aplicaciones de interés y, en cualquier caso, las fuentes originales pueden recuperarse mediante integración.

Si para la separación de las fuentes se utiliza el algoritmo descrito en el capítulo, en este caso particular, el número mínimo de receptores que debe usarse es de $2N+1$, en el caso sin ruido, y de $2N+2$, en un entorno ruidoso. Es decir, se necesitará un receptor más que en el caso general porque la observación del receptor de referencia se le resta al resto, que son las que utiliza el algoritmo de separación propuesto.

En las diversas simulaciones realizadas, salvo en el apartado dedicado a analizar los resultados del caso particular que se acaba de describir, se analizó el caso en el que se supone que las distintas fuentes sufren distintas atenuaciones en su propagación a los receptores, por ser más general, más realista y más restrictivo, en cuanto a los retardos que se pueden introducir, que el descrito en este apartado. Además tal como se

demuestra en el *apéndice A.2*, de cara a una posible implementación física del sistema de separación, colocando receptores en distintas regiones del espacio, este método no permite la separación de mezclas de más de tres fuentes.

4.7 Ampliaciones y mejoras de la versión básica del algoritmo

4.7.1 Introducción

Tal como se ha visto, en la versión básica del algoritmo se hacía uso de un único filtro auxiliar $h(t)$ para obtener la matriz de autocovarianzas filtrada $R_{xx}^h(0)$. El filtro debe ser elegido de forma correcta, ya que debe cumplir una serie de condiciones específicas: su respuesta impulsiva debe ser nula en cero para cancelar los términos de ruido de la ecuación (4.20). Tal como se verá en el capítulo posterior, dedicado al estudio del comportamiento del algoritmo, una elección inadecuada del filtro hará que las estimaciones realizadas de las fuentes sean mucho peores que las obtenidas cuando el filtro es elegido de forma óptima.

Podemos plantearnos usar, en lugar de un único filtro, varios de ellos con respuestas impulsivas diferentes $h_i(t)$, con $i = 1, \dots, n$, para obtener varias matrices de autocovarianzas filtradas $R_{xx}^{h_i}(0)$, en lugar de una sola. Esto hará que dispongamos de información adicional que nos permita hacer una estima más correcta de la matriz de separación \mathbf{W} y que no tengamos que preocuparnos tanto por la elección óptima del filtro $h(t)$.

4.7.2 Descripción del método de extracción de fuentes usando varias matrices de autocovarianzas

En el método de separación anterior, la matriz de separación viene dada por:

$$\mathbf{W} = \mathbf{V}^T \mathbf{U}^T \quad (4.41)$$

Siendo \mathbf{V} la matriz cuyas filas son los autovectores de la matriz \mathbf{R} , que puede expresarse así:

$$\mathbf{R} = [\mathbf{A}^T \mathbf{U}]^{-1} \mathbf{D}_0^{-1} \mathbf{D}_0^h \mathbf{A}^T \mathbf{U} \quad (4.42)$$

Si al ejecutar el algoritmo sólo quisiéramos separar una de las fuentes originales, lo cual equivale a implementar un sistema de extracción ciega de fuentes (BSE), podríamos tomar una fila de la matriz \mathbf{V} correspondiente a un autovector v_1 y obtener una matriz de separación para la fuente esa fuente, que denotaremos como:

$$\mathbf{W}_1 = v_1^T \mathbf{U}^T \quad (4.43)$$

Al multiplicar las señales recibidas en los sensores por la matriz de separación, se obtendrá la fuente correspondiente al autovalor escogido:

$$\mathbf{s}_1 = \mathbf{W}_1 \mathbf{x} \quad (4.44)$$

En general, no sabremos qué autovector corresponde a cada fuente. Por lo que la fuente extraída puede ser cualquiera de las $2N$ fuentes que hay en las mezclas recibidas.

Como vemos la matriz \mathbf{R} depende del tipo de filtro elegido, si en lugar de utilizar un único filtro, se hace uso de varios filtros distintos $h_i(t)$ con $i = 1, \dots, n$, se obtendrán varias matrices de autocorrelación distintas $R_{xx}^{h_i}(0)$ y como la matriz \mathbf{V} depende ellas, se obtendrán varias matrices \mathbf{V}_i . Las filas de cada una de estas matrices constituyen una estimación distinta de cada uno de los autovectores. Obviamente el valor verdadero de los autovectores será muy similar a cada una de estas estimaciones. Intuitivamente puede pensarse que si se hace algún tipo de promedio con las distintas estimas de los autovectores, el valor obtenido estará más próximo al valor verdadero, o lo que es lo mismo, el error que se cometerá en la estima será menor. Ésta es la base teórica de la mejora del algoritmo propuesta.

En el método propuesto, al ejecutar el algoritmo, sólo se extrae una de las fuentes de interés. Para ello se hacen varias estimaciones del autovector asociado a dicha fuente y, a partir de ellas, se hace un promedio para obtener una estima más correcta del autovector v_1 . Finalmente con la estima de v_1 obtenida, se calcula la matriz de extracción \mathbf{W}_1 asociada a la fuente correspondiente a v_1 .

Para el cálculo del autovector deseado a partir de las matrices $R_{xx}^{h_i}(0)$ que se obtendrán con cada filtro, se hará uso del método de las potencias⁶. Este método nos permite obtener de forma sencilla el autovector asociado al autovalor mayor en valor absoluto⁷, es decir, a la fuente de mayor energía. El método nos permite también calcular de forma directa las filas de las matrices \mathbf{V}_i , correspondientes a la fuente que queremos separar en cada ejecución del algoritmo, puesto que el método devolverá siempre el autovector asociado a la misma fuente, la de mayor energía. Por tanto, no tendremos confusión en el cálculo de los autovectores.

El cálculo de la estimación del autovector asociado al autovalor de la fuente con mayor energía se hará de forma iterativa siguiendo los siguientes pasos:

- 1) Asignamos un valor inicial no nulo al autovector que se va a calcular v_1 .
- 2) Repetimos N veces el siguiente proceso de forma iterativa:
 - 2.1) Calculamos los autovalores asociados a v_1 para cada una de las matrices: $\lambda_i = v_1^H R_{xx}^{h_i}(0) v_1$.
 - 2.2) Obtenemos la estimación de v_1 que vendrá dada por:

$$\bar{r}_{xx_1} = \sum_i \lambda_i^* R_{xx}^{h_i}(0) v_1 \quad (4.45)$$

- 2.3) Normalizamos el valor anterior y obtenemos el valor de v_1 :

⁶ Para una descripción de los fundamentos del método véase el Apéndice A.2

⁷ Se supone que todos los filtros $h(t)$ han sido elegidos de forma que los autovalores de las matrices \mathbf{R} sean distintos.

$$v_1 = \frac{\bar{\Gamma}_{xx_1}}{\|\bar{\Gamma}_{xx_1}\|_2} \quad (4.46)$$

Una vez calculado el valor de v_1 óptimo, se obtendrá la matriz de separación correspondiente a la fuente de mayor energía sabiendo que:

$$\mathbf{W}_1 = v_1^T \mathbf{U}^T \quad (4.47)$$

Y la fuente deseada se obtendrá fácilmente a partir de la matriz de separación:

$$y_1 = \mathbf{W}_1 \mathbf{x} \quad (4.48)$$

Así mediante este método sólo podemos extraer una fuente en cada ejecución del algoritmo (la de mayor energía). Como ventaja, la estimación de la fuente extraída será más precisa que en la versión básica del algoritmo. Si queremos extraer el resto de fuentes de interés, tendremos que idear algún método para eliminar de la mezcla de fuentes la señal que se extrajo anteriormente, para que al volver a ejecutar el algoritmo, se extraiga la segunda fuente de mayor energía y así sucesivamente. Este procedimiento se denomina método de deflación.

4.7.3 El método de deflación

La operación de deflación nos va a permitir eliminar del vector de mezclas la presencia de una fuente que previamente se haya extraído. Con ello vamos a conseguir que al volver a ejecutar el algoritmo, se extraiga la segunda fuente de mayor energía. Combinando la ejecución del algoritmo con la operación de deflación vamos a conseguir separar de la mezcla sólo aquellas fuentes que nos interesen ordenadas de mayor a menor energía. Esto nos va a permitir separar de forma sencilla las fuentes de interés de sus derivadas, puesto que si cumplimos la condición descrita en (4.13), los

autovalores de las fuentes serán mayores que los de las derivadas en valor absoluto. Por tanto, ejecutando N veces el algoritmo, obtendremos las fuentes deseadas.

Mediante la deflación vamos a conseguir implementar un sistema de extracción ciega de fuentes (BSE). Esto va a suponer un ahorro del coste computacional con respecto a la versión básica del algoritmo, ya que en ella se utilizaba un sistema de separación ciega de fuentes (BSS) en el que se extraían $2N$ fuentes.

La operación de deflación se realiza de la siguiente forma⁸:

- 1) En primer lugar, se hace una estimación de la columna de la matriz de mezclas asociada a la fuente que se ha extraído. Esto lo hacemos correlando la salida extraída con las observaciones.

La señal extraída se obtendrá como ya comentamos anteriormente:

$$y_1 = \mathbf{W}_1 \mathbf{x} \quad (4.49)$$

Y la estimación de la columna se obtendrá haciendo:

$$a_j = \frac{\mathbf{x} \cdot y_1^T}{T} \quad (4.50)$$

- 2) Buscamos la matriz de proyección ortogonal \mathbf{P}_{A^\perp} que nos elimina la componente de las observaciones que pertenece al subespacio generado por la columna de la matriz de mezclas estimada. Al multiplicar las observaciones por la matriz \mathbf{P}_{A^\perp} , la componente asociada a la columna de la matriz de mezclas estimada se proyectará sobre el origen de ordenadas. Esta matriz se obtiene de la siguiente forma:

$$\mathbf{P}_{A^\perp} = \mathbf{I}_M - a_j \cdot \text{pinv}(a_j) \quad (4.51)$$

Donde
$$\text{pinv}(a_j) = a_j \cdot (a_j^T a_j)^{-1} a_j^T$$

⁸ Para una descripción de las bases matemáticas en las que se inspira la operación de deflación, véase [13]

- 3) Multiplicando \mathbf{P}_{A^\perp} por las observaciones se obtendrá la deflación de la fuente extraída:

$$\mathbf{x}_{\text{nuevo}} = \mathbf{P}_{A^\perp} \cdot \mathbf{x} \quad (4.52)$$

Tras realizar la operación de deflación, se habrá eliminado de la mezcla la presencia de la señal que hemos extraído. Para volver a extraer las siguientes fuentes, sólo tendremos que volver repetir el proceso de ejecución del algoritmo seguido de la deflación de la última fuente estimada.

4.8 Resumen del capítulo

En el presente capítulo se ha propuesto y se ha justificado la validez de un método para la separación de una mezcla de fuentes recibida en un conjunto de sensores y obtenida a partir de la mezcla de un conjunto de fuentes estadísticamente independientes entre sí y emitidas desde distintos puntos. Las mezclas recibidas en los sensores estarán formadas por una combinación lineal de las fuentes originales con un cierto retardo, pues se asume que el medio introduce un pequeño retardo en la propagación de las fuentes.

Además se han propuesto ciertas mejoras en el algoritmo para solventar algunas de sus carencias y obtener una mayor precisión de las estimaciones realizadas.

También se ha descrito un método para la optimización del comportamiento del algoritmo de separación frente a los retrasos cuando se cumplen ciertas condiciones. Esto se consigue operando con las observaciones recibidas.

El método de separación propuesto se implementó en MATLAB. En capítulos posteriores, se realizarán varias simulaciones para estudiar el comportamiento del algoritmo y la validez de las suposiciones y las aproximaciones que se han realizado en este capítulo.

CAPÍTULO 5

SIMULACIONES Y ESTUDIO DEL COMPORTAMIENTO DEL ALGORITMO

5.1 Ilustración del funcionamiento de la versión básica del algoritmo

5.1.1 Introducción

El algoritmo propuesto en el capítulo anterior para la separación de fuentes con pequeños retardos se implementó en el lenguaje de programación MATLAB con el objeto de estudiar el correcto funcionamiento del mismo. En este capítulo, se analizará el comportamiento del algoritmo y la validez de las aproximaciones realizadas en el capítulo anterior. Además se presentarán los resultados obtenidos en las simulaciones de aquellos esquemas de separación que se han considerado más significativos, para mostrar cómo se comporta el método de separación propuesto, al variar los diversos parámetros que intervienen en el proceso de mezcla y separación de las fuentes.

5.1.2 Ejemplo de separación de mezclas de fuentes con retardos

Para mostrar el funcionamiento del algoritmo, se procedió a separar una mezcla de cuatro fuentes de audio procedentes de cuatro hablantes distintos.

Para la separación se emplearon un total de $M = 9$ receptores. Según se vio en el capítulo anterior, el número de receptores debe ser, en general, mayor que $2N$ siendo, en este caso, $N = 4$.

La información recibida en cada uno de los receptores consistía en una combinación lineal de las fuentes transmitidas con un cierto retraso en cada uno de ellos. Se asume que las cuatro fuentes están en la misma región del espacio y que los receptores se encuentran en una misma región del espacio, aunque separados una pequeña distancia entre sí, por lo que las señales serán recibidas en ellos con distintos retardos porque la distancia de cada fuente a cada receptor es distinta.

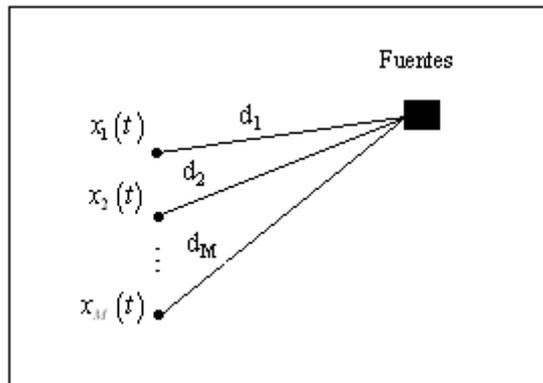


Figura 5.1: Situación de las fuentes y de los receptores.

Para generar la información recibida en los receptores, se introdujeron en cada receptor unos retardos distintos y aleatorios en las fuentes originales y posteriormente se mezclaron las fuentes retardadas con una matriz de mezclas de dimensión $N \times 1$. La matriz de mezclas se genera de forma aleatoria en cada simulación y sus elementos son números reales uniformemente distribuidos en el intervalo $[-1,1]$.

Según lo visto en el tema anterior, para que el retardo desde la fuente j hasta el receptor i , que denotamos como τ_{ij} se considere pequeño debe verificarse:

$$\tau_{ij}^2 \ll \frac{1}{2\pi^2 f_{\max}^2} \quad \forall i, j \quad (5.1)$$

Donde f_{\max} es el ancho de banda máximo de las señales que se transmiten.

Puesto que se trabaja con señales de audio, se escogió una $f_{\max} = 2\text{KHz}$. Se tomó este valor para respetar la condición anterior y a la vez permitir que las señales de audio con las que se trabaja sean perfectamente inteligibles, a fin de poder comprobar fácilmente su calidad tras el proceso de separación.

Para el filtrado de las señales de audio, se utilizó un filtro LP elíptico. Se utilizó un filtro elíptico porque es el que necesita menor orden para satisfacer unas especificaciones dadas. Sus características son las siguientes:

- Extremo de la banda de paso $f_p = 2\text{KHz}$
- Extremo de la banda de rechazo $f_s = 2.08\text{KHz}$
- Rizado en la banda de paso $R_p = 0.5\text{dB}$
- Rizado en la banda de rechazo $R_s = 30\text{dB}$

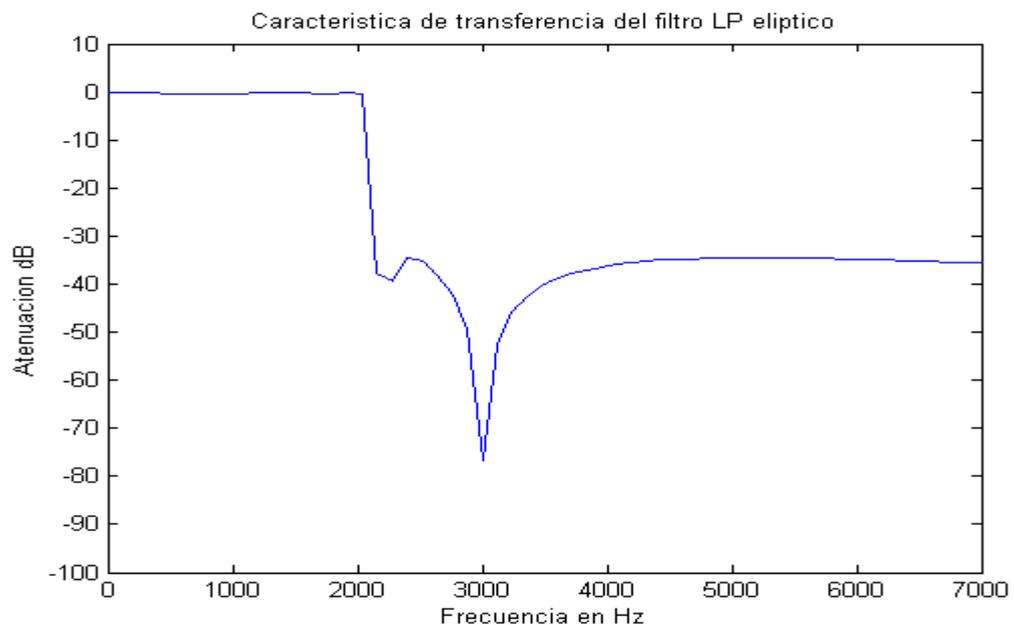


Figura 5.2: Característica de transferencia del filtro LP utilizado

Si en la desigualdad (5.1) imponemos que el término de la izquierda sea un orden de magnitud menor que el de la derecha, se obtendrá la siguiente ecuación:

$$10\tau_{ij}^2 = \frac{1}{2\pi^2 f_{\max}^2} \quad (5.2)$$

Sustituyendo f_{\max} , se obtiene el retraso máximo admisible que podemos tener:

$$\tau_{ij} = \frac{1}{\sqrt{20\pi}f_{\max}} = 3.6 \cdot 10^{-5} s \quad (5.3)$$

Las señales de audio utilizadas fueron grabadas con una tasa de muestreo de 48KHz. Se escogió una tasa de muestreo elevada para obtener un margen más amplio en los retardos que se pueden introducir y a la vez cumplir la condición de pequeños retardos.

Así, con los valores que se tomaron el número máximo de muestras que podemos retrasar cumpliendo la desigualdad (5.1) es:

$$3.6 \cdot 10^{-5} s \times \frac{48000 \text{muestras}}{s} = 1.75 \approx 2 \text{muestras} \quad (5.4)$$

En la simulación se asignaron retardos a las fuentes de una o dos muestras a las fuentes de forma arbitraria. Dichos retardos se introdujeron filtrando las fuentes con un filtro de respuesta impulsiva $h(t) = \delta(t - n_0)$, siendo n_0 el número de muestras que se decidió retardar.

Por último a las señales recibidas en los receptores se les añadió ruido gaussiano⁹ de media cero, de forma que se obtuvo una relación señal a ruido aproximada de 30dB.

A continuación se presentan las señales originales, sus mezclas retardadas y las estimaciones de las señales originales que realiza el algoritmo:

⁹ Posteriormente se estudiará el comportamiento del algoritmo frente al ruido.

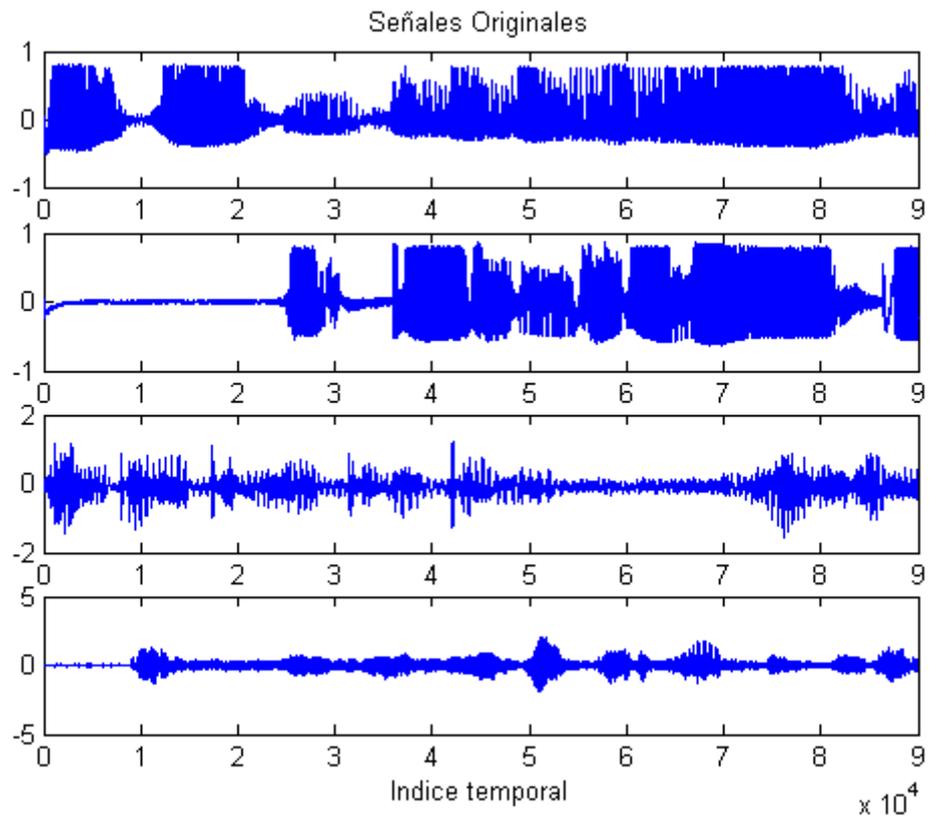


Figura 5.3: Señales de audio emitidas por las fuentes

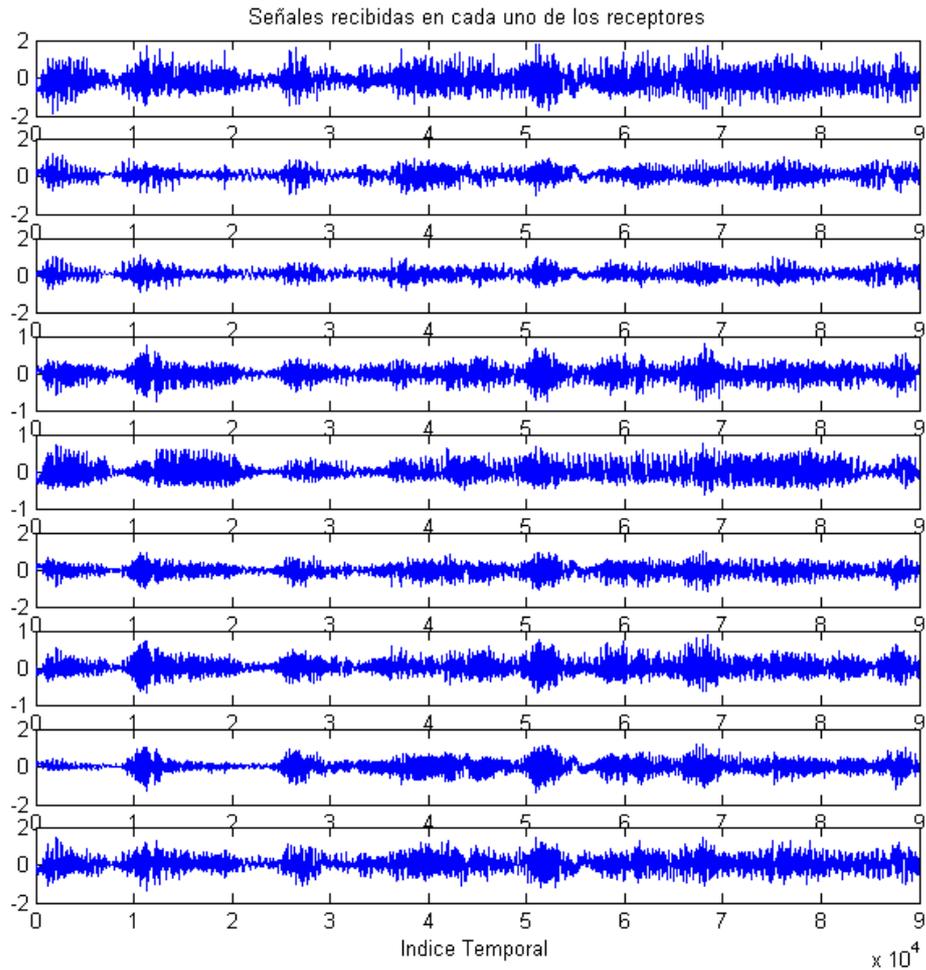


Figura 5.4: Mezclas de señales recibidas en los receptores

Tal como puede verse en la *figura 5.4*, en cada receptor se recibe una combinación lineal de las fuentes originales con un cierto retardo, más el ruido de tipo gaussiano que se ha añadido para hacer modelar con más precisión las señales que se reciben en los sensores. Tal como se vio en el capítulo anterior, las señales recibidas en los receptores pueden expresarse de la forma:

$$x_i(t) = \sum_{j=1}^N c_{ij} s_j(t - \tau_{ij}) + n_i(t) \quad (5.5)$$

Con $i = 1, \dots, M$.

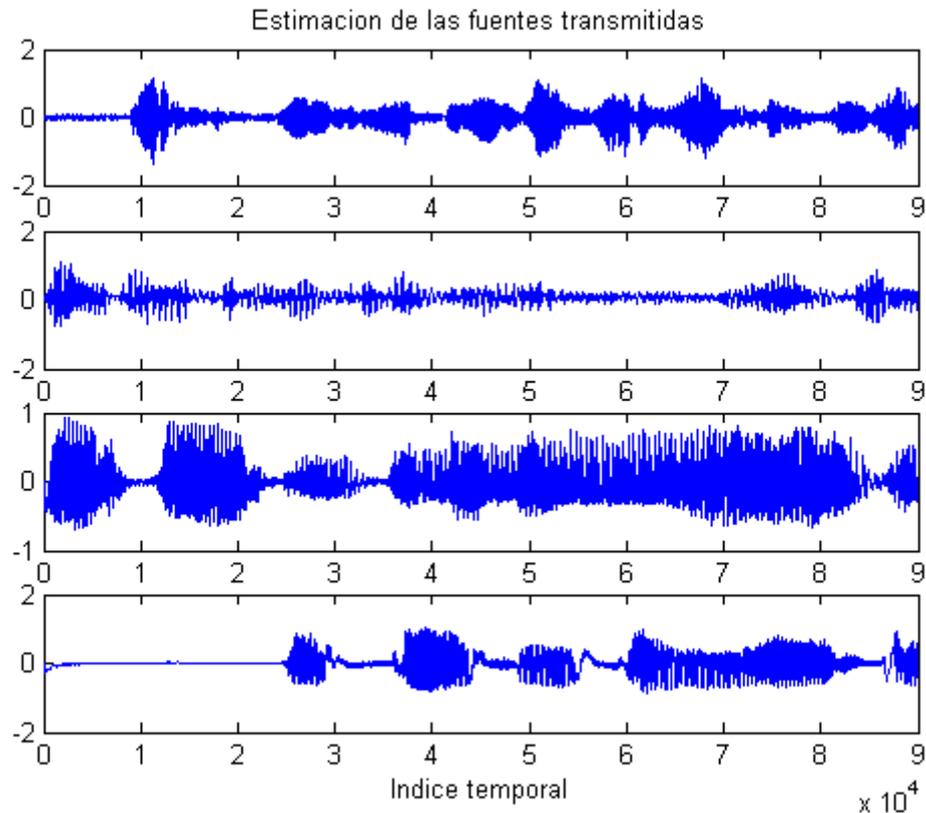


Figura 5.5: Estimación de las fuentes que se transmitieron

En la *figura 5.5*, se aprecia la estimación realizada por el algoritmo de las fuentes originales a partir de las mezclas. Las señales obtenidas son bastante parecidas a las originales. A simple vista, puede asociarse cada estima con la fuente correspondiente. También puede verse que el algoritmo es incapaz el orden y la amplitud de las señales que se mezclaron, tal como se comentó en el capítulo 2 en el apartado dedicado a las restricciones de ICA.

5.1.3 Obtención de las derivadas de las fuentes originales

Según el estudio realizado en el apartado anterior, la señal recibida en el receptor *i*-ésimo puede escribirse de la siguiente forma:

$$x_i(t) = \sum_{j=1}^N c_{ij} s_j(t - \tau_{ij}) + n_i(t) \quad (5.6)$$

Teniendo en cuenta que los términos de la forma $s(t-\tau)$ pueden expresarse de manera aproximada como $s(t-\tau) \approx s(t) - \tau \dot{s}(t)$, la señal recibida en el receptor viene dada por:

$$x_i(t) \approx c_{i,1}s_1(t) - c_{i,1}\tau_{i,1} \frac{ds_1(t)}{dt} + c_{i,2}s_2(t) - c_{i,2}\tau_{i,2} \frac{ds_2(t)}{dt} + \dots \quad (5.7)$$

$$\dots + c_{i,N}s_N(t) - c_{i,N}\tau_{i,N} \frac{ds_N(t)}{dt} + n_i(t)$$

El algoritmo de separación propuesto considera las derivadas primeras de las fuentes transmitidas como nuevas fuentes a separar. Tal como puede verse en la expresión anterior, dichas derivadas aparecen multiplicadas por el retardo τ_{ij} que se produce desde la fuente j hasta el receptor i . Este término de retardo es, por definición, pequeño y hará que la amplitud de las derivadas de las fuentes sea varios órdenes de magnitud más pequeña que la amplitud de las fuentes originales. En el caso particular de la simulación realizada, el valor máximo admisible que se obtuvo para el retardo desde cualquier fuente a cualquier receptor fue de $\tau_{ij_{\max}} = 3.6 \cdot 10^{-5} s$.

Es de esperar que, siempre que se cumpla la condición de pequeños retardos, la estimación que realiza el algoritmo de las derivadas no sea adecuada, puesto que su amplitud es mucho menor que la de las fuentes originales. Además si existe ruido, puede ocurrir que la amplitud del mismo sea del mismo orden de magnitud que la de las derivadas o incluso superior, por lo que la tarea de estimación de las derivadas se dificulta aún más. Cuando no se cumpla la desigualdad (5.1), la estimación de las derivadas será correcta puesto que su amplitud será comparable a la de las fuentes.

De cualquier manera, el objetivo del algoritmo es estimar las fuentes originales y no sus derivadas. Si se quisiera hacer una estimación de las mismas, se obtendrían resultados mucho más exactos si simplemente se calculara la derivada primera de las estimaciones de las fuentes.

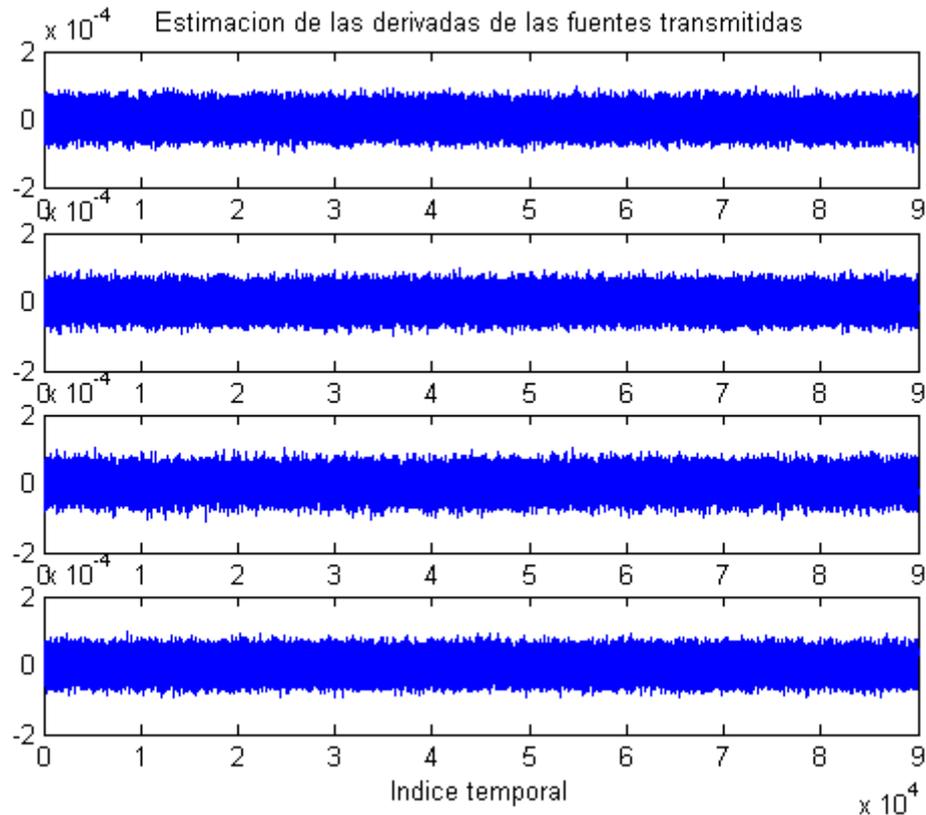


Figura 5.6: Estimación de las derivadas primeras de las fuentes.

Tal como puede verse en las gráficas de arriba, el algoritmo es incapaz de determinar con exactitud las derivadas en condiciones normales de funcionamiento, ya que éstas aparecen enmascaradas por el propio ruido que se ha introducido en la simulación, ya que su energía es menor que la del ruido. Esta será la situación que se tendrá siempre que se introduzca un nivel de ruido apreciable.

Para mostrar la estimación que el algoritmo hace de las derivadas, se realizó una simulación en las mismas condiciones del apartado anterior, pero con una salvedad, se eliminó el ruido (condiciones ideales) para que éste no enmascare a las derivadas.

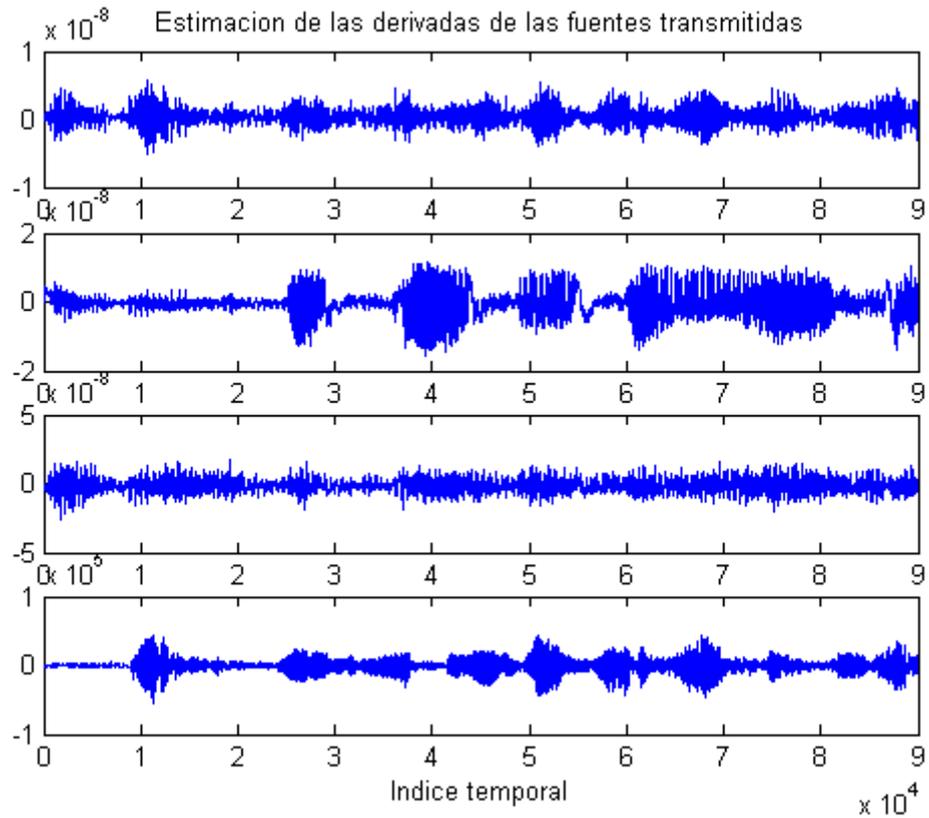


Figura 5.7: Estimación realizada de las derivadas en ausencia de ruido

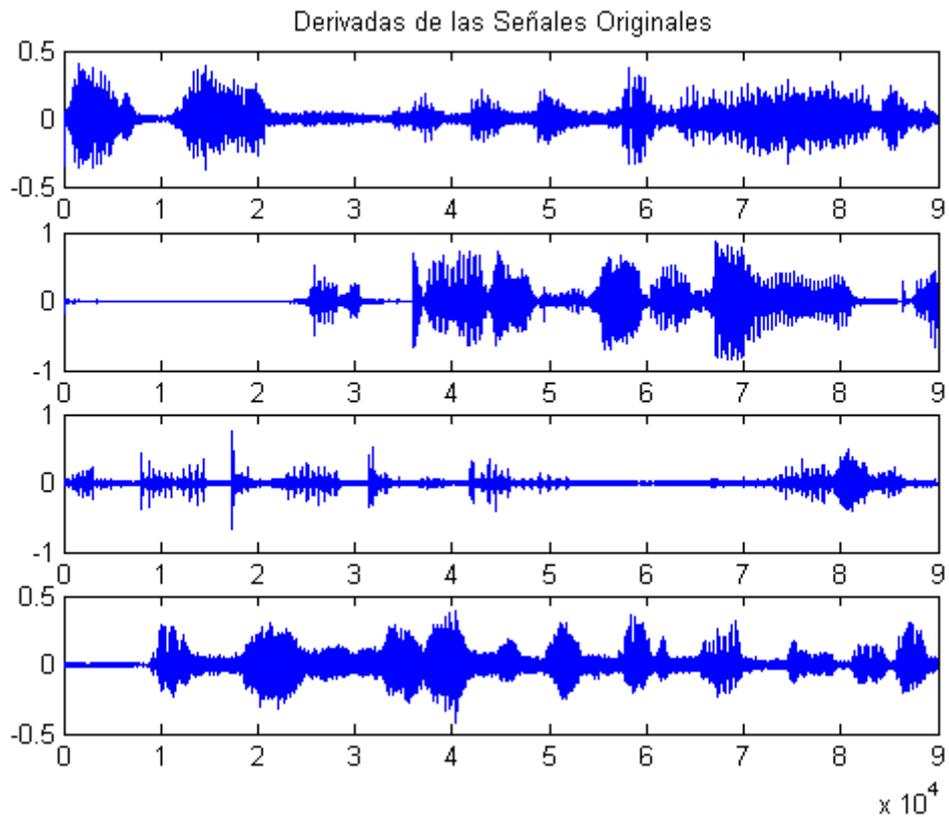


Figura 5.8: Valores reales de las derivadas de las fuentes transmitidas

Comparando las estimaciones de las derivadas con y sin ruido, puede verse que mejora de forma notable cuando se elimina el ruido. Pese a todo, si se comparan las estimaciones realizadas de las derivadas con respecto a las derivadas de cada una de las fuentes, vemos que la estimación realizada por el algoritmo sigue siendo bastante burda.

A pesar de ello, es posible identificar a simple vista a qué derivada corresponde cada una de las derivadas estimadas por el algoritmo. La primera gráfica de la *figura 5.7* es la estimación de la primera gráfica de la *figura 5.8* y así sucesivamente.

En resumen, la estimación de las derivadas que hace el algoritmo no es buena debido a la pequeña amplitud de éstas frente a la de las fuentes. Esto es algo lógico, ya que si unas señales poseen mucha más energía que otras, es habitual que los algoritmos ICA fallen. Pese a todo, esto no supone un problema puesto que el objetivo del algoritmo no es la obtención de las derivadas, sino la de las fuentes originales.

En general, el algoritmo devuelve las señales estimadas (derivadas y fuentes) desordenadas. La separación puede hacer simplemente, escogiendo las N señales de mayor energía. Estas se corresponderán con las señales de interés.

Posteriormente se verá cómo las ampliaciones realizadas sobre el algoritmo básico permiten obtener de manera directa las fuentes de interés, sin tener que hacer una separación entre fuentes y derivadas.

También se mostrará cómo al introducir retardos por encima de los admisibles, la estima de las derivadas se hace de forma correcta, ya que estos casos la amplitud de las derivadas no es despreciable frente a la de las fuentes y a la del propio ruido.

5.1.4 Comportamiento del algoritmo frente al ruido

5.1.4.1 Respuesta ante el ruido

Tal como se introdujo en el capítulo anterior, supondremos que el ruido recibido en todos los receptores es blanco y gaussiano. Por simplicidad, se supondrá que el ruido recibido en todos los receptores tiene la misma varianza σ^2 .

El cálculo de la relación señal a ruido en cada receptor es complejo. Si a cada receptor se le asocia una fuente de interés, en él se recibe además del ruido y de la propia señal de interés, una combinación lineal del resto de señales. Esta combinación lineal puede considerarse como un ruido adicional o interferencia.

En las simulaciones se emplearon señales de audio con un nivel de energía similar. A la hora de estimar la relación señal a ruido, se utilizó el siguiente criterio:

$$SNR = \frac{E_{señales}}{E_{ruido}} \quad (5.8)$$

Siendo $E_{señales}$ la energía de las señales y E_{ruido} la energía de la señal de ruido que se introdujo. Es un criterio que arroja resultados pesimistas porque en cada receptor se recibe ruido adicional debido a combinaciones lineales de señales distintas a la de interés, que pueden considerarse como interferencia, por lo que la SNR será más baja en realidad que los resultados que nos dé la expresión anterior.

Para evaluar la robustez del algoritmo frente al ruido, se realizaron varias simulaciones con un número de fuentes comprendido entre 3 y 5. En ellas, se fue aumentando de forma progresiva el nivel de ruido para observar las degradaciones producidas en las señales estimadas¹⁰.

- Para relaciones señal a ruido $SNR > 18dB$ el comportamiento del algoritmo es bastante acertado y las señales de audio son extraídas sin que se perciba la influencia del ruido. Para relaciones en el rango $14dB < SNR < 18dB$, las señales son extraídas de forma correcta, sin embargo, al oírlas puede apreciarse la influencia de un cierto nivel de ruido de fondo.
- Si bajamos de una $SNR = 14dB$, el algoritmo dejará de comportarse correctamente y en las señales comenzarán a apreciarse pequeños lóbulos debido a las otras señales distintas a la de interés. Por lo que al escuchar las señales, se oirán fuentes superpuestas a la de interés.
- Para $SNR < 10dB$, el algoritmo deja de funcionar y en las estimas se obtiene una superposición de las señales de interés.

¹⁰ En dichas simulaciones se eligió el filtro y el número de receptores de forma óptima

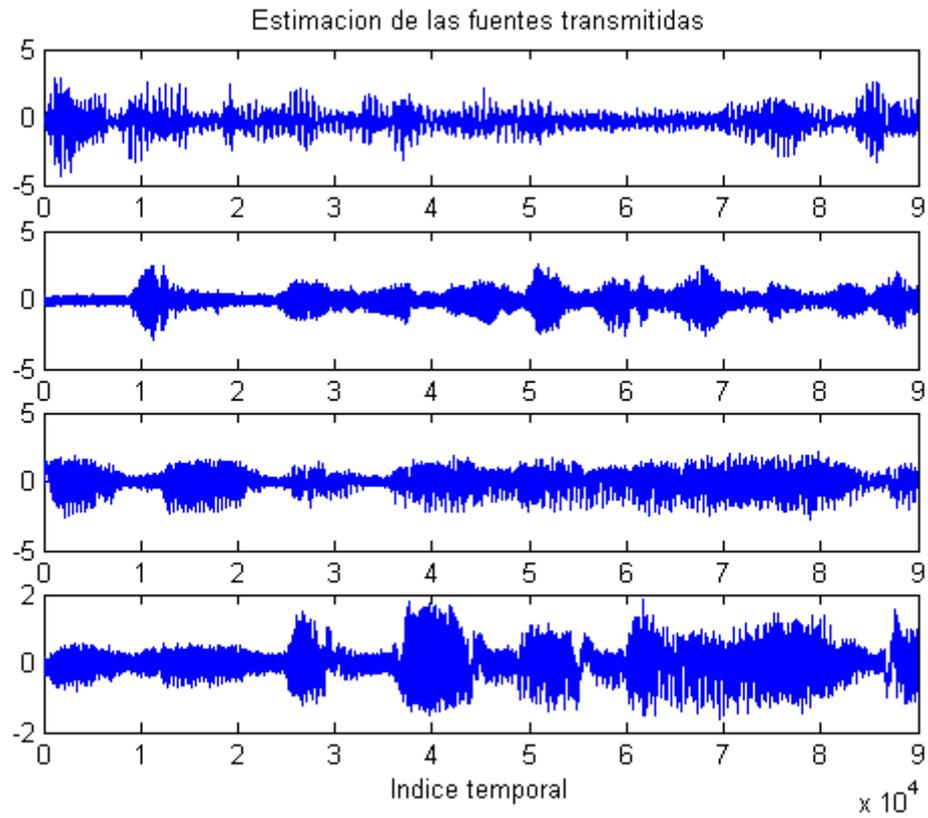


Figura 5.9: Estimaciones de las señales para una $SNR \approx 8dB$

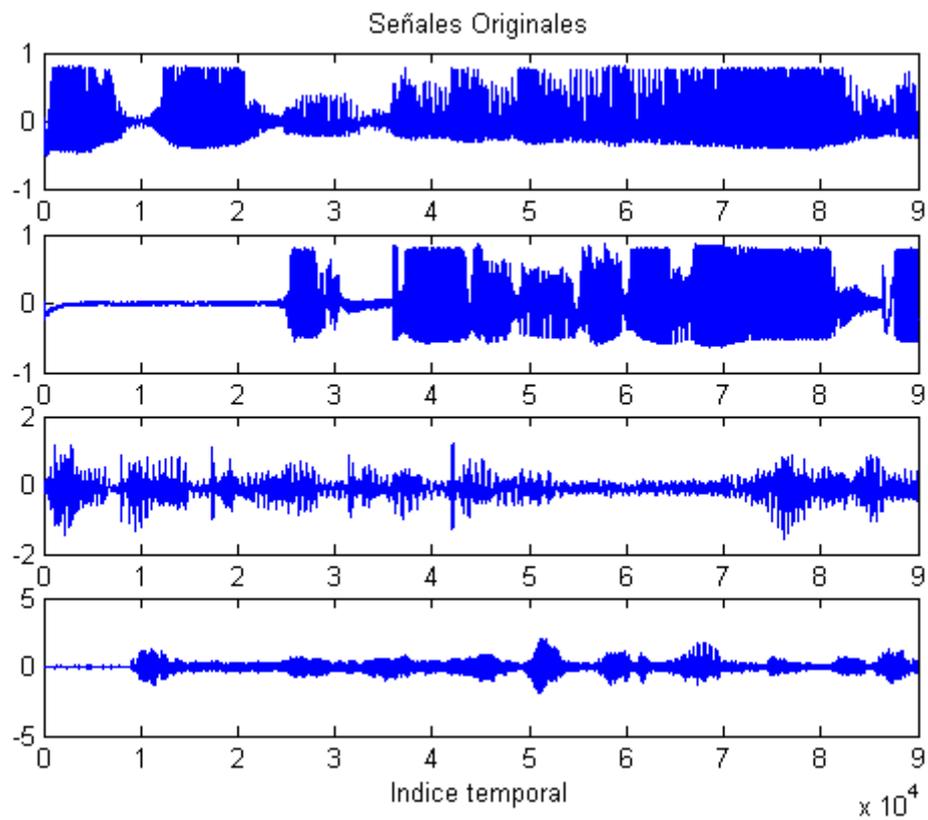


Figura 5.10: Señales originales

En las figuras anteriores puede verse lo que ocurre cuando empiezan a introducirse relaciones señal a ruido por debajo de los $10dB$, al intentar extraer las fuentes originales, se obtienen combinaciones lineales de éstas porque el algoritmo confunde el espacio de las fuentes originales con el del ruido, ya que cuando su energía se hace comparable a la de las fuentes, puede considerarse el ruido como una nueva fuente estadísticamente independiente del resto.

5.1.4.2 Elección del filtro utilizado para la obtención de la segunda matriz de autocovarianzas

Según vimos en el capítulo anterior el algoritmo de separación se basa en el uso de dos matrices de autocovarianzas para la obtención de la matriz de separación. La matriz de autocovarianzas de las observaciones y la matriz de autocovarianzas de las observaciones filtrada con un filtro con respuesta impulsiva $h(t)$:

$$\mathbf{R}_{xx}(0) = \mathbf{A}\mathbf{D}_0\mathbf{A}^T + \sigma^2\mathbf{I}_M \quad (5.9)$$

$$\mathbf{R}_{xx}^h(0) = \mathbf{A}\mathbf{D}_0^h\mathbf{A}^T + \sigma^2h(0)\mathbf{I}_M \quad (5.10)$$

Para facilitar resolver el problema según el método propuesto el filtro ha de escogerse de forma que su respuesta impulsiva sea nula en cero, de forma que las ecuaciones anteriores queden así:

$$\mathbf{R}_{xx}(0) = \mathbf{A}\mathbf{D}_0\mathbf{A}^T + \sigma^2\mathbf{I}_M \quad (5.11)$$

$$\mathbf{R}_{xx}^h(0) = \mathbf{A}\mathbf{D}_0^h\mathbf{A}^T \quad (5.12)$$

En principio, cualquier filtro no nulo, con respuesta impulsiva nula en cero, será válido para que el algoritmo funcione. Sin embargo, el comportamiento frente al ruido depende mucho del filtro que se elija. Una correcta elección del filtro hará mucho más robusto al algoritmo frente al ruido.

Una elección incorrecta del filtro hará que la relación señal a ruido mínima para la cual el efecto del ruido es imperceptible aumente de forma considerable. Así en las

simulaciones realizadas con un filtro adecuado, se obtuvo una relación señal a ruido mínima 20dB más pequeña que cuando el filtro se elige de forma arbitraria.

El filtro $h(t)$ escogido para todas las simulaciones es el siguiente:

$$h(t) = \delta(t-1) + \delta(t-2) + \dots + \delta(t-n_0) \quad (5.13)$$

Siendo n_0 un entero comprendido en el rango $[10,15]$. El filtro presenta respuesta impulsiva nula en cero para cancelar el término debido al ruido y el resto de sus coeficientes son iguales a la unidad. En las señales de voz, las distintas muestras están parcialmente correladas para valores o distancias de hasta unas 15 muestras. Se trata de escoger un filtro que altere lo menos posible las propiedades de la señal de audio.

5.1.4.3 Comportamiento del algoritmo al variar el número de receptores

Según lo visto en el capítulo anterior, el número mínimo de receptores que debe emplearse es de $M = 2N$ en el caso de que no haya ruido y de $M = 2N + 1$ en el caso con ruido. Nos centraremos, a partir de ahora, en este último caso por ser el más general.

Tal como se vio, el algoritmo para calcular la matriz de separación realiza la descomposición en valores singulares de la matriz de autocovarianzas de las observaciones:

$$\mathbf{R}_{xx}(0) = [\mathbf{u}_1, \dots, \mathbf{u}_M] \mathbf{L} [\mathbf{u}_1, \dots, \mathbf{u}_M]^T \quad (5.14)$$

$$\text{Siendo } \mathbf{L} = \text{diag}(\lambda_1^2 + \sigma^2, \dots, \lambda_{2N}^2 + \sigma^2, \sigma^2, \dots, \sigma^2) \quad (5.15)$$

Y a partir de la media de los últimos $M - 2N$ elementos de la diagonal de la matriz \mathbf{L} se realiza una estimación de la varianza del ruido σ^2 . Lógicamente mientras más próxima sea esta estimación al valor real mejor será el comportamiento del algoritmo. Por tanto, es de esperar que si se aumenta el número de receptores, los resultados dados por el algoritmo mejoren en presencia de ruido.

Los resultados obtenidos utilizando un número adecuado de filtros son mucho mejores que si se utiliza el número mínimo permitido.

Para mostrar el comportamiento frente al ruido al variar el número de receptores, se realizó una simulación con tres fuentes y se analizó la calidad de las estimaciones del algoritmo usando el número mínimo de receptores $M_1 = 2N + 1$ y un número suficientemente elevado (se tomó $M_2 = 14$).

En las simulaciones se obtuvo que la relación señal a ruido mínima para la que el ruido es imperceptible con un número adecuado de receptores es de $SNR_{\min} = 18dB$, mientras que si el número de receptores se reduce al mínimo, $SNR_{\min} = 28dB$, por lo que tomando un número suficientemente elevado de receptores, se obtiene una mejora de $10dB$.

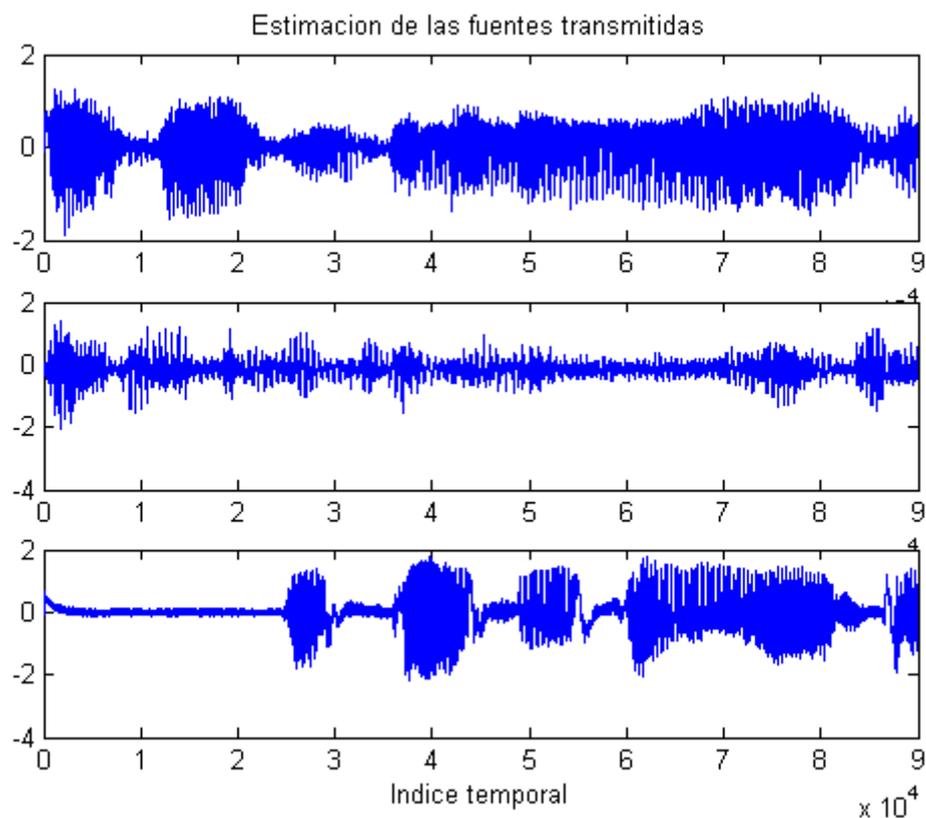


Figura 5.11: Estimación de las fuentes con una $SNR \approx 12dB$ y $M_2 = 14$

En la figura de arriba puede verse la estimación de las fuentes realizada por el algoritmo para una relación señal a ruido de $12dB$. Si comparamos las estimaciones de

las fuentes con las fuentes originales de la *figura 5.3*, vemos que el algoritmo se comporta de forma correcta.

A continuación se redujo el número de receptores al mínimo. Tal como puede verse en la gráfica de abajo, el algoritmo es incapaz de estimar correctamente las fuentes que se transmitieron.

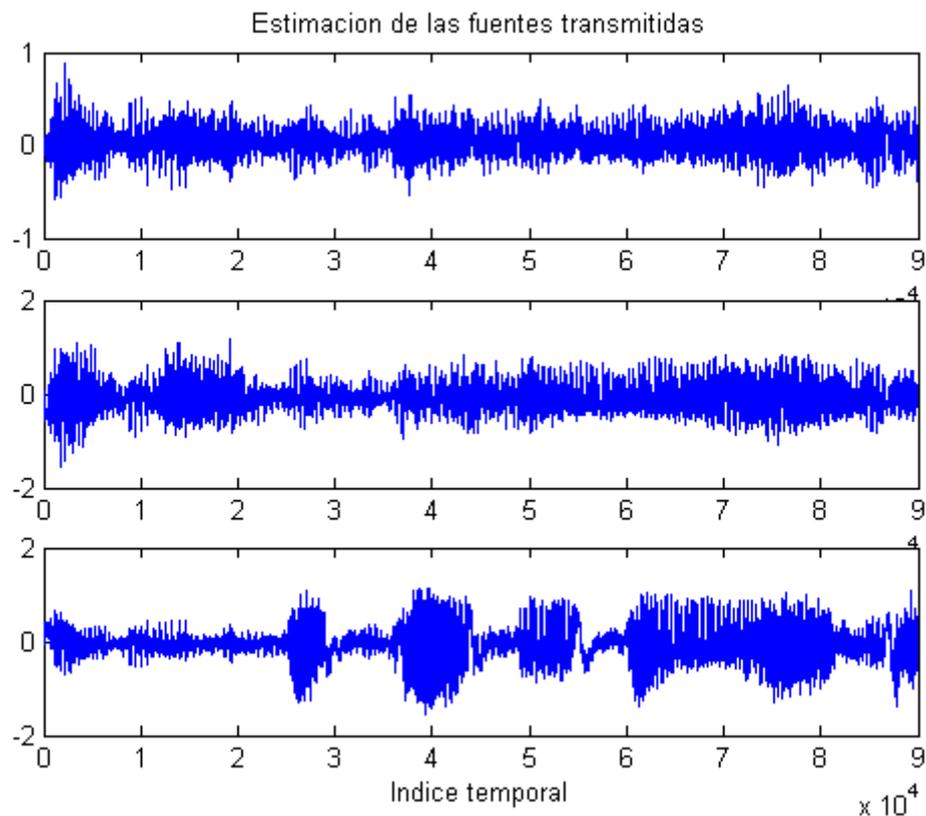


Figura 5.12: Estimación de las fuentes transmitidas para una $SNR \approx 12dB$ y $M_1 = 7$

En la práctica, al aumentar el número de receptores por encima de un cierto umbral, no se obtienen mejoras en el comportamiento del algoritmo.

A la vista de los resultados obtenidos en varias simulaciones, un número bastante acertado para el número de receptores resultó ser el doble del número mínimo que podemos tener, es decir, $M = 2*(2N + 1)$.

Es preciso remarcar que el algoritmo maneja matrices de hasta dimensión $M \times M$, por lo que si se aumenta mucho el número de receptores, porque queramos mejorar el comportamiento ante el ruido o porque el número de fuentes a separar sea elevado, el tiempo de respuesta aumentará considerablemente.

5.1.5 Comportamiento frente a los retardos de las fuentes

Para mostrar el comportamiento del algoritmo frente a los retardos, se realizaron varias simulaciones con las mismas condiciones que en el apartado 5.1.2. Tal como se estableció anteriormente, de acuerdo con la ecuación (5.2), el número máximo de retardos admisibles entre las fuentes recibidas es de dos muestras. Para retardos mayores, se estará violando la condición de pequeños retardos. En este apartado, se hicieron simulaciones introduciendo retardos mayores que los admisibles y se estudió el comportamiento del método de separación en estos casos. Los resultados obtenidos fueron los siguientes:

- Para retardos de 1 a 2 muestras (tal como se mostró) el comportamiento es el esperado.
- Cuando se introducen retardos de 3 a 4 muestras (ya estamos fuera de los límites establecidos) las fuentes se estiman de forma correcta también. En el proceso de extracción, ya empiezan a estimarse de forma correcta algunas de las derivadas de las fuentes. Ello se debe a que en la ecuación (5.7) los términos correspondientes a las derivadas ya no son despreciables frente a las fuentes porque los términos debidos a los retrasos τ_{ij} han aumentado considerablemente.
- Para retrasos en torno a las 8 muestras (las suposiciones realizadas ya no son correctas) el algoritmo deja de funcionar de forma adecuada. Las fuentes extraídas aunque todavía se reconocen bien, tienen una amplitud bastante pequeña comparable a la de las propias derivadas. En estos casos, ya la estima de las derivadas es correcta, aunque en algunos casos el algoritmo omite alguna de ellas, especialmente, si el nivel de ruido que se introduce es elevado. Decir que bajo estas condiciones, el algoritmo se vuelve mucho más sensible al ruido. Para relaciones señal a ruido por debajo de los $20dB$, el algoritmo ya no es capaz de estimar de forma correcta las fuentes.

- Para retrasos mayores de las 10 muestras, el comportamiento es inaceptable. La amplitud de las fuentes estimadas es muy baja. Además éstas aparecen distorsionadas y muy afectadas por el ruido.

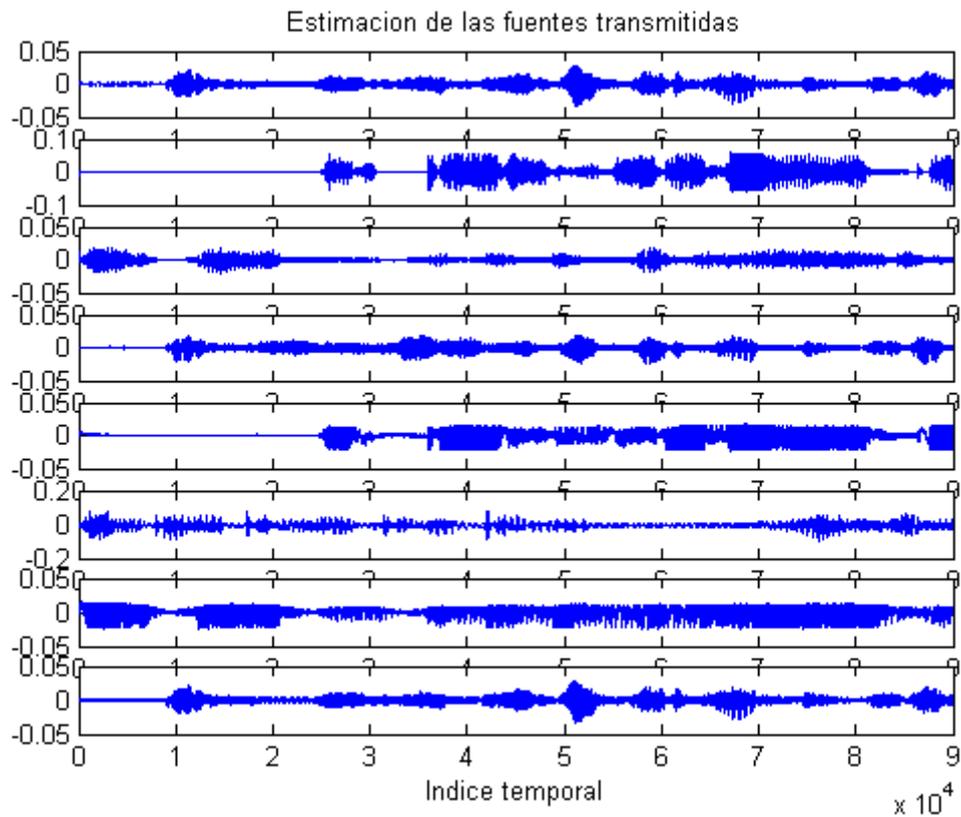


Figura 5.13: Estimación de las fuentes y sus derivadas para retrasos 8 muestras y una $SNR \approx 30dB$

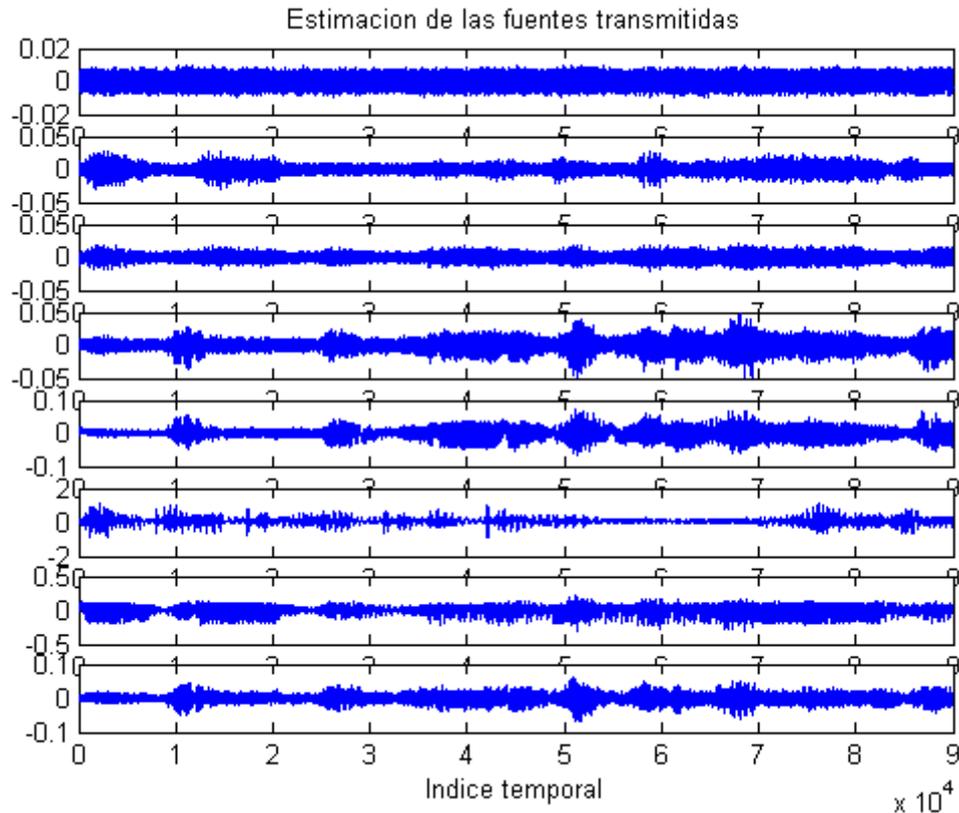


Figura 5.14: Estimación de las fuentes y sus derivadas para retrasos en torno a 11 muestras y una $SNR \approx 30dB$

En la *figura 5.13*, puede verse la estimación de las fuentes originales y sus derivadas cuando se introducen retardos de hasta 8 muestras. Es fácil identificar a simple vista las fuentes obtenidas, si las comparamos con las señales que aparecen en la *figura 5.3* (fuentes originales) y las de la *figura 5.8* (derivadas primeras de las fuentes originales).

Para estos retardos tan elevados la amplitud de las fuentes deseadas se hace muy pequeña. Además su calidad es baja debido a la distorsión.

En la *figura 5.14* se observan los resultados obtenidos para retrasos de hasta 11 muestras. Claramente el algoritmo no funciona bien y tanto las fuentes como sus derivadas no se estiman de forma correcta.

Un caso particular en el que también podemos tener retardos, es el de la *figura 5.15*. Se trata de la situación contraria considerada hasta ahora, las fuentes se encuentran separadas una pequeña distancia entre sí y los receptores se encuentran juntos en una misma región del espacio.

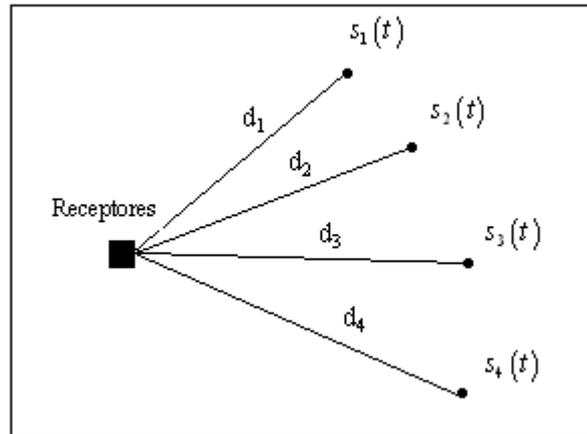


Figura 5.15: Segundo esquema de simulación posible

Las señales recibidas en los receptores pueden expresarse matemáticamente de la forma:

$$x_i(t) = \sum_{j=1}^N c_{ij} s_j(t - \tau_j) + n_i(t) \quad (5.16)$$

En este caso, las señales de interés son recibidas en cada uno de los receptores con el mismo retardo puesto que se consideran que la distancia que los separa es nula. Podemos considerar el problema de separación como un problema de separación de mezclas instantáneas en el que las fuentes de interés son las señales retrasadas. Es por ello, que al simular esta situación, el algoritmo se mostró inmune a los retrasos introducidos. Se introdujeron retrasos muy por encima de los límites admisibles y la separación de las fuentes siempre fue perfecta. Por ello, en las diversas simulaciones realizadas se simuló la situación propuesta en la *figura 5.1*, que es la más restrictiva frente a los retardos y la que se propone en el artículo original, en el que se describe el algoritmo, véase [1].

Un tercer esquema de simulación más complejo consiste en una combinación de los dos anteriores, es decir, tanto las fuentes como los receptores se encuentran separados una pequeña distancia entre sí. En este caso, el comportamiento del algoritmo frente a los retardos fue exactamente el mismo que el obtenido con el primer esquema de simulación.

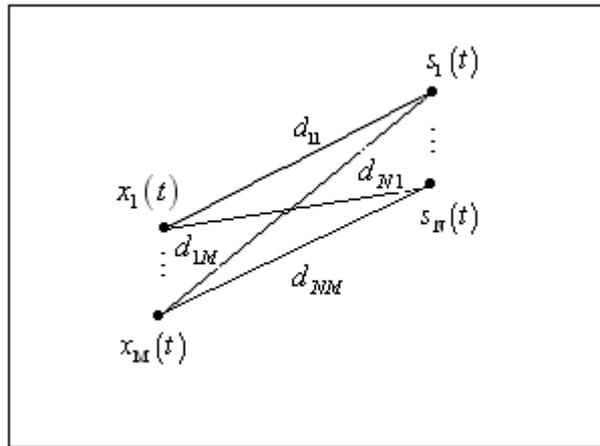


Figura 5.16: Tercer esquema de simulación posible

A modo de resumen, podemos decir que el comportamiento del algoritmo frente a los retardos es el esperado y que las hipótesis realizadas en el capítulo anterior a cerca de los retardos se cumplen. Además se ha visto que para retardos de hasta 4 muestras (el doble del valor establecido para los retardos máximos admisibles) el algoritmo sigue funcionando correctamente. A la vista de esto, si en la desigualdad (5.1) imponemos que el término de la izquierda sea unas cinco veces mayor que el de la derecha, en lugar de un orden de magnitud como se estableció inicialmente, obtendremos una expresión empírica para el retardo máximo que se puede introducir, acorde con las simulaciones realizadas. Así el retardo máximo admisible puede determinarse de forma aproximada mediante la siguiente expresión¹¹:

$$\tau_{ij_{\max}} \approx \frac{1}{\sqrt{10\pi} f_{\max}} \quad (5.17)$$

¹¹ Para relaciones señal a ruido por encima de los 20dB

5.2 Estudio del funcionamiento de la versión ampliada del algoritmo

5.2.1 Introducción

En este apartado se presentarán los resultados obtenidos en las diversas simulaciones de la versión ampliada del algoritmo que se expuso en el capítulo anterior. Dicha ampliación consistía básicamente en el uso de varios filtros en lugar de uno solo y en el uso de la operación de deflación para extraer las fuentes una a una.

A la vez que se vayan presentando los resultados obtenidos, se irán haciendo comparaciones con los resultados obtenidos con la otra versión del algoritmo, con el fin de evaluar las prestaciones de cada una de las dos implementaciones.

5.2.2 Ejemplo de separación de fuentes con retardos

Para mostrar el funcionamiento de la versión ampliada del algoritmo, se realizó una simulación en las mismas condiciones que en el apartado 5.1.2. Al igual que antes, se pretendía separar las voces de cuatro hablantes distintos, $N = 4$, usando un total de $M = 9$ receptores.

Las voces utilizadas fueron las mismas que en el apartado anterior. Además se filtraron de la misma forma que antes utilizando el filtro con la característica de transferencia presentada en la *figura 5.2* y el nivel de ruido que se introdujo no se varió.

Los retrasos se introdujeron de la misma forma que en la simulación anterior. Al igual que antes, se obtuvieron unos retardos máximos admisibles de dos muestras entre las señales recibidas.

La novedad de esta simulación consistió en el uso de varios filtros, en lugar de uno, para mejorar la estimación de las fuentes. Se decidió utilizar un total de cuatro filtros¹², $N_{Fil} = 4$. Los filtros utilizados presentaban todos respuesta impulsiva nula en cero, tenían una longitud de unas 15 muestras y presentaban pequeñas variaciones entre

¹² Posteriormente se presentará un análisis del comportamiento frente al número de filtros.

ellos. Básicamente estos filtros eran secuencias formadas por un número inicial de ceros pequeño en las primeras muestras, seguidos por un número mayor de unos en el resto de muestras. La razón de la elección de este tipo de filtros es que fueron los que mostraron mejor comportamiento por separado en la versión básica del algoritmo.

A continuación en la *figura 5.17*, se presenta la estimación realizada de las fuentes originales. Si se comparan a simple vista con las fuentes originales de la *figura 5.18*, apenas se observan diferencias entre las estimaciones y las fuentes reales.

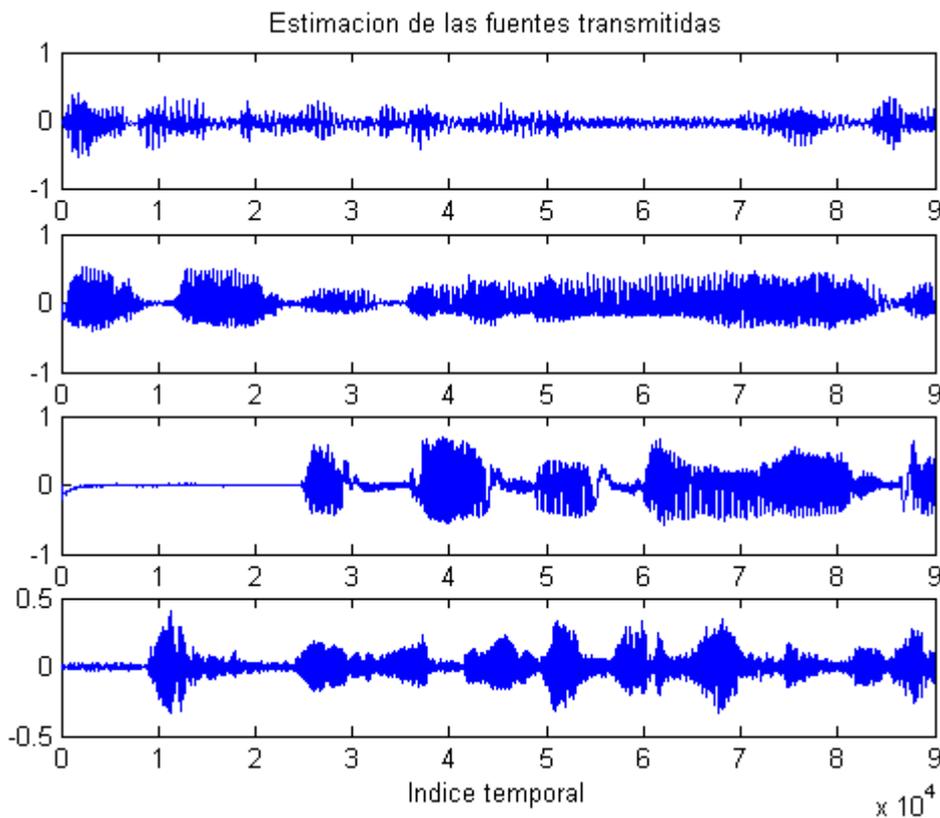


Figura 5.17: Estimación de las fuentes transmitidas en la versión mejorada del algoritmo

Como novedad, al ejecutar el algoritmo dentro de los límites admisibles para los retardos, las N primeras fuentes que se extraigan se corresponderán con las señales de interés, por lo que ya no tendremos que preocuparnos por la identificación de las fuentes y las derivadas al ejecutar el algoritmo, como ocurría en el caso anterior. Además, si decidimos extraer sólo las fuentes de interés, obtendremos un ahorro en el coste computacional. Pese a ello, esta versión es mucho más lenta que la anterior porque todos los cálculos matriciales con los vectores de observación deben repetirse cada

vez que se realiza la operación de deflación y, en este caso, se realizan cuatro veces más que en el ejemplo anterior.

Si se comparan los resultados obtenidos por las dos versiones del algoritmo en las mismas condiciones, observando los resultados obtenidos, a simple vista no se aprecian diferencias, ya que ambas versiones funcionan de forma acertada dentro de los límites establecidos. Tan sólo se aprecian algunas pequeñas diferencias al escuchar las señales extraídas. En la versión ampliada, la calidad de las señales de audio es mejor, ya que la influencia del ruido que se añadió apenas se nota.

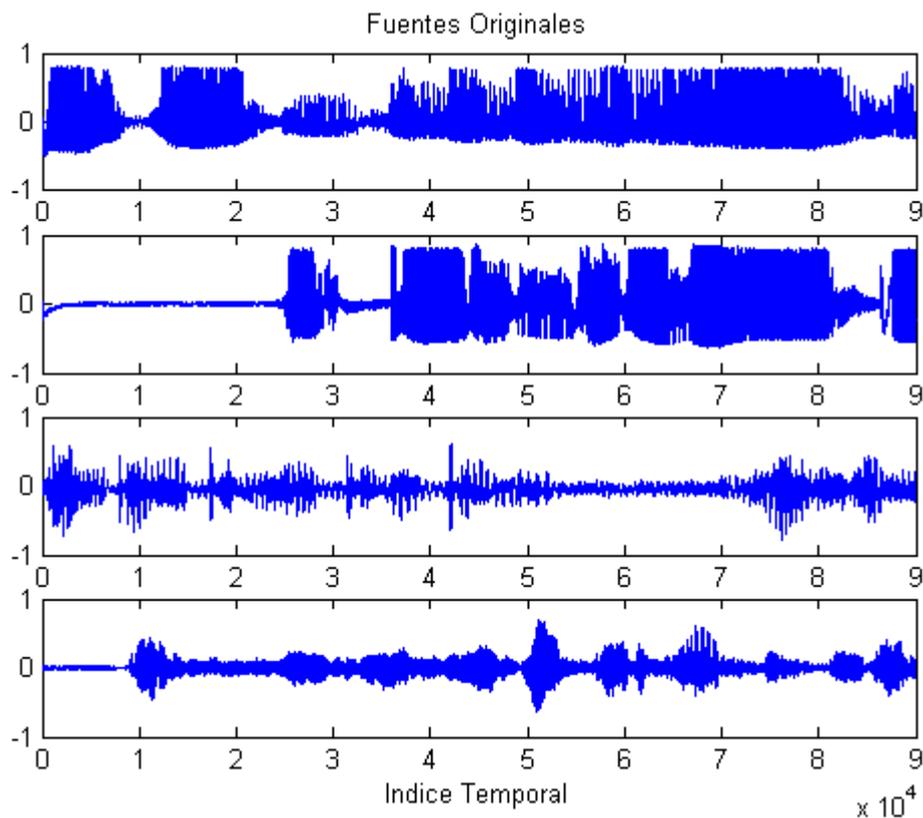


Figura 5.18: Fuentes Originales

5.2.3 Obtención de las derivadas de las fuentes originales

Al igual que en la versión anterior, la estimación de las derivadas, dentro de los límites admisibles para los retrasos, no es buena. Tal como se demostró anteriormente, las señales recibidas en el *i-ésimo* receptor pueden expresarse de la forma:

$$\begin{aligned}
 x_i(t) \approx & c_{i,1}s_1(t) - c_{i,1}\tau_{i,1} \frac{ds_1(t)}{dt} + c_{i,2}s_2(t) - c_{i,2}\tau_{i,2} \frac{ds_2(t)}{dt} + \dots \\
 & \dots + c_{i,N}s_N(t) - c_{i,N}\tau_{i,N} \frac{ds_N(t)}{dt} + n_i(t)
 \end{aligned} \tag{5.18}$$

Si los retardos son pequeños (del orden de microsegundos en la simulación anterior), la amplitud de las fuentes originales será varios órdenes de magnitud menor que la de las derivadas, lo cual dificulta el buen funcionamiento del algoritmo, ya que la operación de deflación siempre deja una parte residual de la señal que queremos eliminar en la mezcla. Si las derivadas son muy pequeñas, estas porciones residuales de las señales de interés que previamente se han extraído y que aún permanecen en la mezcla pueden ser dominantes sobre ellas, lo cual hará que la estimación sea peor que en la versión básica del algoritmo. Además, si se añade ruido, tal como se ha hecho en todas las simulaciones, en muchos casos la energía de la señal de ruido puede ser mayor que la de las derivadas, por lo que el algoritmo confundirá el espacio de las derivadas con el de la señal de ruido.

Si los retardos introducidos están dentro de los límites, la operación de deflación nos permitirá discernir entre las derivadas y las fuentes de interés. Las N primeras fuentes extraídas serán las de interés y las últimas N fuentes se corresponderán con las derivadas. Generalmente sólo se extraerán las N primeras fuentes, puesto que las derivadas no interesan.

En el caso de que se introduzcan retardos muy por encima de los admisibles, es conveniente realizar la extracción de las $2N$ fuentes, puesto que la derivada de alguna fuente puede hacerse mayor que otra fuente distinta y, si sólo se extraen las N primeras fuentes, puede perderse la estimación de alguna de las señales de interés porque en su lugar se extraiga la derivada de otra fuente.

5.2.4 Comportamiento frente al ruido

5.2.4.1 Respuesta frente al ruido

Para analizar el comportamiento de la versión ampliada del algoritmo, se realizaron varias simulaciones en las que se fue variando la relación señal a ruido. Los

retardos introducidos estaban dentro de los límites admisibles y se mantuvieron constantes en las diversas simulaciones.

Al igual que en el estudio que se hizo del comportamiento frente al ruido en la versión básica del algoritmo, en las simulaciones se introdujo ruido blanco y gaussiano. Además, el nivel de ruido que se introdujo en todos los receptores fue el mismo, de varianza σ^2 , pues ésta es una de las hipótesis en las que se fundamenta el método de separación propuesto.

La estimación de la relación señal a ruido se hizo de la misma forma que en el apartado 5.1.4.1, la SNR se determinó de la siguiente forma:

$$SNR = \frac{E_{señales}}{E_{ruido}} \quad (5.19)$$

Siendo $E_{señales}$ el nivel de energía de las fuentes, que, por defecto, es el mismo para todas, y E_{ruido} es la energía de la señal de ruido introducida. Tal como se argumentó en el apartado 5.1.4.1, es una estimación que arroja resultados pesimistas. Además, en este caso, este hecho se acentúa debido a que la operación de deflación siempre deja un residuo, de la señal que queremos extraer, en la mezcla. Estas señales residuales que quedan en la mezcla, se comportarán como ruido adicional que dificultará la extracción de las señales siguientes.

Las simulaciones se realizaron, al igual que en la versión básica, con un número de fuentes comprendido entre 3 y 5 y se fue variando el número de filtros usados. En ellas se fue disminuyendo el nivel la SNR de forma progresiva para ver la robustez del algoritmo.

En general, los resultados obtenidos fueron similares a los de la versión básica del algoritmo. Pese al uso de varios filtros, la relación señal a ruido mínima a partir de la cual el algoritmo dejaba de funcionar correctamente, resultó ser aproximadamente la misma que en la versión básica. Sin embargo, la calidad de las señales de audio estimadas en presencia de ruido resultó ser mayor, ya que el efecto del ruido aparece mucho más mitigado. Esto es lógico, pues si se hace uso de varios filtros y éstos han sido elegidos de forma adecuada, la estimación de la matriz de separación debe ser más precisa. Los resultados obtenidos en las diversas simulaciones, al ir variando la SNR , fueron los siguientes:

- Para relaciones señal a ruido $SNR > 17dB$, el comportamiento es bastante acertado y las señales de audio son extraídas sin que se aprecie la influencia del ruido.

- Para relaciones en el rango $14dB < SNR < 17dB$, las señales son extraídas de forma correcta, sin embargo, al oírlas puede apreciarse la influencia de un cierto nivel de ruido de fondo. En esta versión del algoritmo, la influencia que se percibe del ruido es menor que en la anterior.

Para relaciones señal a ruido menores de 17dB, la operación de deflación puede fallar en alguna de las iteraciones del algoritmo, por lo que el algoritmo puede devolver dos o más estimaciones consecutivas de una misma señal. Si esto ocurre, ya las N primeras estimaciones no se corresponderán con las señales de interés y será preciso examinar todas las $2N$ fuentes estimadas. La identificación de las señales de interés será fácil, puesto que las N fuentes de interés se extraen correctamente, aunque no en orden.

- Si bajamos de una $SNR = 14dB$, el algoritmo dejará de comportarse correctamente y en las señales comenzarán a apreciarse pequeños lóbulos debido a las otras señales distintas a la de interés. Por lo que al escuchar las señales extraídas, se oirán fuentes superpuestas. Al igual que antes, puede ocurrir que algunas señales de interés estimadas no aparezca entre las N primeras fuentes devueltas por el algoritmo.

- Para $SNR < 10dB$, el algoritmo deja de funcionar y en las estimas se obtiene una superposición de las señales de interés.

A continuación se muestran los resultados obtenidos en una simulación con una $SNR = 15dB$, tres fuentes y tres filtros distintos. Para estas relaciones señal a ruido relativamente bajas, a veces, la operación de deflación falla en alguna de las iteraciones del algoritmo, tal como se muestra en la *figura 5.19*. Este fenómeno se verá acentuado

si una de las fuentes es claramente dominante frente al resto, es decir, su energía es mucho mayor que la de las demás.

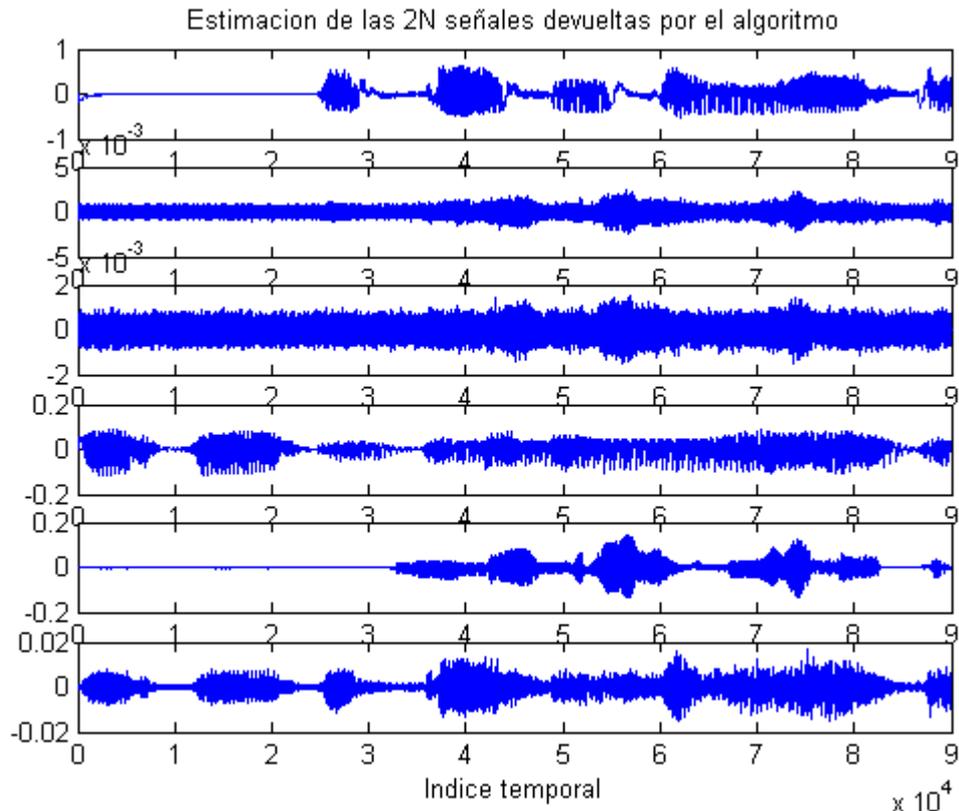


Figura 5.19: Salidas del algoritmo obtenidas al mezclar tres fuentes con una $SNR \approx 15dB$ y usando tres filtros distintos.

Si se observan las estimaciones realizadas por el algoritmo y se comparan con las fuentes originales mostradas en la *figura 5.20*, puede verse a simple vista que la primera estimación se corresponde con la segunda fuente, la cuarta estimación con la primera y la quinta estimación con la tercera fuente. Las N primeras estimaciones no se corresponden con las fuentes originales porque cuando se trabaja con relaciones señal a ruido bajas, a veces, la operación de deflación falla. En la *figura 5.19* puede verse cómo la operación de deflación de la primera fuente no se realiza de forma correcta dos veces consecutivas. Una vez que ésta se ha realizado completamente, el algoritmo continúa con la extracción del resto de fuentes, que ahora sí se realiza correctamente.

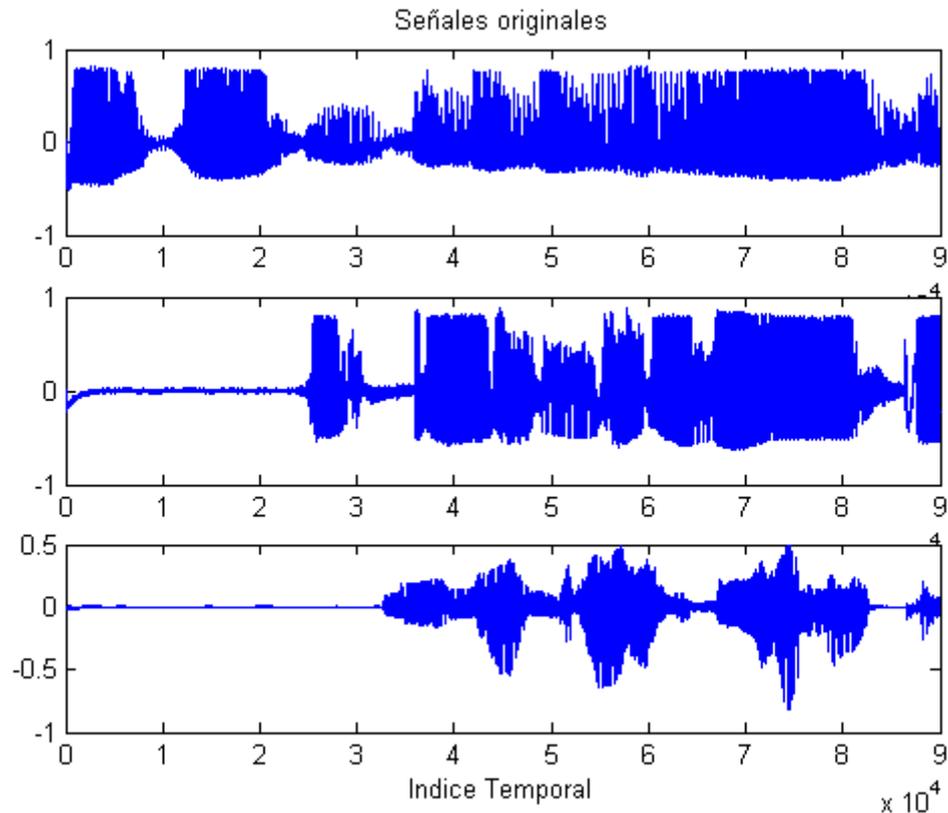


Figura 5.20: Fuentes originales utilizadas para la simulación de la *figura 5.19*

5.2.4.2 Comportamiento del algoritmo al variar el número de receptores

Los resultados obtenidos en este apartado fueron similares a los obtenidos en la versión básica del algoritmo. Tal como se comentó anteriormente, en general, la calidad de las señales extraídas mejora si se aumenta el número de receptores, puesto que la estimación de la varianza del ruido es más precisa. El número mínimo de receptores que puede usarse en presencia de ruido es de $M = 2N + 1$, por lo que habrá de elegirse un número mayor o igual a éste en las simulaciones.

En esta versión del algoritmo, es especialmente importante la elección correcta del número de receptores. Eligiéndolo de forma adecuada, puede obtenerse una mejora significativa del comportamiento del algoritmo frente al ruido. En el caso de que el número de receptores no se elija correctamente, tanto por exceso como por defecto, el rendimiento del algoritmo se verá mermado.

Si el número de receptores es escaso, el algoritmo se volverá mucho menos robusto frente al ruido. Al igual que en la versión básica, en las simulaciones se obtuvo que la

relación señal a ruido mínima para la que el ruido es imperceptible, con un número adecuado de receptores, es de $SNR_{\min} = 17dB$, mientras que si el número de receptores se reduce al mínimo, $SNR_{\min} = 27dB$, por lo que tomando un número suficientemente elevado de receptores, se obtiene una mejora de $10dB$.

Si el número de receptores es excesivo, el tiempo de respuesta del algoritmo aumentará de forma vertiginosa. Esto se debe a que para obtener las matrices de autocorrelación filtradas correspondientes a cada filtro, hemos de filtrar cada una de las filas de la matriz de autocorrelación de las observaciones $\mathbf{R}_{xx}(0)$ de dimensión $M \times M$ con el filtro que corresponda. Lógicamente si se hace uso de un número de filtros elevado, hemos de cuidar no aumentar excesivamente el número de receptores.

Además cuando el número de receptores es muy elevado, del orden $4N$, la operación de deflación, a veces, no se realiza de forma correcta, si una de las fuentes es dominante frente al resto, ya que si la señal no se elimina correctamente de la señal de mezcla de alguno de los receptores, puede ocurrir que el algoritmo realice dos extracciones de la misma señal.

A partir de las diversas simulaciones que se realizaron, un valor adecuado para el número de receptores resultó ser $M = M_{\min} + 2$, siendo $M_{\min} = 2N + 1$.

5.2.5 Elección de los filtros usados y estudio del comportamiento del algoritmo al variar el número de filtros

En esta implementación del algoritmo, se hace uso de varios filtros elegidos por el usuario para la obtención de varias matrices de autocovarianzas filtradas, en lugar de una como en la versión anterior. Cada una de estas matrices de autocovarianzas puede expresarse de la forma:

$$\mathbf{R}_{xx}^{h_i}(0) = \mathbf{A}\mathbf{D}_0^{h_i}\mathbf{A}^T + \sigma^2 h_i(0)\mathbf{I}_M \quad \text{Con } i = 1, \dots, N_{Fil} \quad (5.20)$$

Siendo $h_i(t)$ la respuesta impulsiva de cada uno de los filtros utilizados. Tal como se argumentó anteriormente, la respuesta impulsiva de los filtros debe ser nula en cero y así poder expresar la ecuación anterior de la forma:

$$\mathbf{R}_{xx}^{h_i}(0) = \mathbf{A}\mathbf{D}_0^{h_i}\mathbf{A}^T \quad (5.21)$$

En la versión anterior del algoritmo, la elección correcta del filtro a utilizar era clave. Así en las simulaciones realizadas se obtuvo que eligiendo el filtro de forma adecuada, puede obtenerse una relación señal a ruido mínima, para la que el ruido es imperceptible, del orden de $20dB$ menor que si el filtro se elige de forma arbitraria.

En esta versión la dependencia con los tipos de filtros escogidos no es tan grande. Si se escogen varios filtros de forma correcta y otros de forma arbitraria, el algoritmo devolverá resultados aceptables. Esto es así porque, tal como se explicó anteriormente, el algoritmo realiza un promedio entre todos los autovectores correspondientes a una misma fuente, que han sido obtenidos mediante el uso de diversas matrices de autocovarianzas filtradas. Por tanto, si alguno de los filtros se elige de forma incorrecta, no influirá tan negativamente como en la versión anterior.

Los filtros $h_i(t)$ escogidos en las simulaciones fueron similares al filtro utilizado en la versión básica del algoritmo, ya que de todos los filtros que se probaron, fue el que mejor comportamiento tuvo. Dicho filtro tenía una respuesta impulsiva de la forma:

$$h(t) = \delta(t-1) + \delta(t-2) + \dots + \delta(t-n_0) \quad (5.22)$$

Siendo n_0 un entero comprendido en el rango $[10,15]$.

Los filtros $h_i(t)$ utilizados y que dieron mejores resultados fueron conjuntos de filtros como el utilizado en la versión básica, con pequeñas diferencias, como retardos de algunas muestras o variaciones de su longitud en algunas muestras.

Con el propósito de determinar el número adecuado de filtros a utilizar, se realizaron varias simulaciones con una $SNR \approx 20dB$ en las que se fueron variando el número de filtros usados. El número de filtros que se fue incrementando desde 1 hasta 8. Todos los filtros que se utilizaron cumplían con los requisitos necesarios y fueron probados por separado para comprobar que su comportamiento era adecuado.

También se realizaron varias simulaciones en las mismas condiciones anteriores en las que se utilizó un conjunto de filtros elegidos de forma adecuada y uno o dos filtros con un funcionamiento inadecuado.

A partir de las diversas simulaciones, se obtuvieron las siguientes conclusiones:

- En general, si los filtros utilizados se han elegido correctamente, la calidad de las señales de audio es algo mayor que en la versión con un solo filtro, aunque estas diferencias no son demasiado significativas.
- Al aumentar el número de filtros, el comportamiento del algoritmo mejora si éstos han sido elegidos de forma correcta. Para conjuntos de filtros mayores de tres, las diferencias en el comportamiento no son significativas.
- Es preferible hacer uso de número reducido de filtros (uno o dos), que hayan sido elegidos de forma óptima, a hacer uso de un número elevado de filtros entre los que haya algunos con comportamiento mediocre.
- El tiempo de respuesta del algoritmo aumenta de forma considerable a medida que se introducen filtros, puesto que los procesos de filtrado son los que más recursos consumen.

5.2.6 Comportamiento frente a los retardos de las fuentes

Para estudiar el comportamiento frente al retardo, se realizaron varias simulaciones en las que se fue variando el número de retardos que sufrían las fuentes en su trayecto hacia los receptores. Se consideró la situación descrita en la *figura 5.1*, en la que las distintas fuentes procedentes desde un foco puntual se propagan hacia los receptores, separados pequeñas distancias entre sí. En la simulación se emplearon un total del $M = 8$ receptores para separar mezclas de $N = 3$ fuentes. Se utilizó una $SNR = 30dB$, para que el ruido no influyera mucho en el proceso de separación de las fuentes. El número de filtros utilizados fue de $N_{Fil} = 3$, todos ellos elegidos de forma adecuada. Por último, el resto de parámetros se mantuvieron de la misma forma que en el apartado 5.1.2.

Tal como se estableció en dicho apartado, los retardos máximos admisibles, bajo las condiciones que en él se especifican, son de 2 muestras según la condición (5.2).

A modo de resumen, los resultados obtenidos en las simulaciones fueron los siguientes:

- Para retrasos de hasta dos muestras (estamos dentro de los límites admisibles) el comportamiento del algoritmo fue el esperado.
- Para retrasos de tres muestras (ya estamos por encima de los límites del algoritmo) el comportamiento es bueno, aunque, a veces, las fuentes de interés no aparecen entre las N primeras devueltas por el algoritmo porque la operación de deflación no se realiza de forma correcta.
- Para retrasos desde 5 hasta 10 muestras, las fuentes son extraídas de forma correcta, sin embargo, a veces, el algoritmo omite alguna de ellas y en muchos casos, el orden de las estimaciones no es el esperado.
- A partir de retrasos de 13 muestras, el algoritmo deja de funcionar y sólo se obtienen mezclas de las fuentes de interés. A veces, el algoritmo realiza correctamente la estima de una fuente (la de mayor energía).
- Para los otros dos esquemas de simulación propuestos, los resultados obtenidos fueron los mismos que con la versión básica.

A partir de los resultados obtenidos, podemos decir que el comportamiento frente a los retardos es similar al de la versión básica. Al igual antes, la condición para los retardos máximos admisibles dada por (5.2) resultó ser demasiado restrictiva. Una expresión más próxima a la realidad para los retardos máximos que se pueden introducir, es la dada por la expresión empírica que se determinó en el apartado anterior:

$$\tau_{j_{\max}} \approx \frac{1}{\sqrt{10\pi} f_{\max}} \quad (5.23)$$

5.3 Comparación entre las dos versiones del algoritmo

A la vista de los resultados obtenidos en las simulaciones, el comportamiento de ambas versiones es similar.

La versión ampliada del algoritmo proporciona señales con algo de más calidad y no exige una elección óptima del filtro para proporcionar resultados aceptables. Además tiene la ventaja de permitirnos extraer sólo las señales de interés. Como desventaja su coste computacional es mucho más elevado.

La versión básica presenta resultados bastante similares a los de la versión ampliada si el filtro se escoge de forma adecuada y su tiempo de respuesta es mucho menor.

Comparando ambas versiones, puede decirse que la versión básica presenta resultados bastante similares a los de la versión avanzada, si el filtro se elige de forma adecuada (esto no supondrá un problema, puesto que ya se han descrito los tipos de filtro que proporcionan mejores resultados). Además los resultados son estimados en mucho menor tiempo, lo cual es una gran ventaja de cara a una posible implementación física de un sistema de separación en tiempo real que haga uso del algoritmo. La gran desventaja de la versión básica es que devuelve las señales de interés y sus derivadas mezcladas. La amplitud de las derivadas es despreciable frente a la de las fuentes, por lo que pueden idearse métodos basados en la energía de las señales para separar unas de otras. Por tanto, de cara a una posible implementación física, la versión básica, con una pequeña ampliación que nos permita separar las fuentes originales de sus derivadas, quizás sea la opción más acertada.

5.4 Resultados obtenidos al utilizar la expansión en serie de Taylor de segundo orden

5.4.1 Introducción

Tal como se expuso en el capítulo anterior, en el caso particular de que todas las fuentes sufran la misma atenuación en su propagación hacia los receptores, es posible realizar una optimización en el algoritmo, operando con las señales recibidas, que nos permite introducir retrasos mayores de los impuestos por la condición (5.1).

En el capítulo anterior, vimos que las observaciones en los diversos receptores distintos al de referencia con $i = 2, \dots, M$ pueden expresarse con respecto a la observación del sensor de referencia, con $i = 1$, de la forma:

$$x_i(t) \approx s_1(t) - \tau_{i,1} \frac{ds_1(t)}{dt} + \frac{\tau_{i,1}^2}{2} \frac{d^2s_1(t)}{dt^2} + s_2(t) - \tau_{i,2} \frac{ds_2(t)}{dt} + \frac{\tau_{i,2}^2}{2} \frac{d^2s_2(t)}{dt^2} + \dots \quad (5.24)$$

$$\dots + s_N(t) - \tau_{i,N} \frac{ds_N(t)}{dt} + \frac{\tau_{i,N}^2}{2} \frac{d^2s_N(t)}{dt^2} + n_i(t)$$

Y la observación en el sensor de referencia, $i = 1$, es igual a:

$$x_1(t) = s_1(t) + s_2(t) + \dots + s_N(t) + n_1(t) \quad (5.25)$$

Si a las observaciones en los receptores distintos al de referencia les restamos las observaciones del de referencia, tendremos $M - 1$ observaciones, con $i = 2, \dots, M$, cuyo valor aproximado es:

$$\tilde{x}_i(t) \approx -\tau_{i,1} \frac{ds_1(t)}{dt} + \frac{\tau_{i,1}^2}{2} \frac{d^2s_1(t)}{dt^2} - \tau_{i,2} \frac{ds_2(t)}{dt} + \frac{\tau_{i,2}^2}{2} \frac{d^2s_2(t)}{dt^2} + \dots \quad (5.26)$$

$$\dots - \tau_{i,N} \frac{ds_N(t)}{dt} + \frac{\tau_{i,N}^2}{2} \frac{d^2s_N(t)}{dt^2} + n_i(t)$$

La idea es aplicar el algoritmo de separación a las $M-1$ observaciones anteriores para obtener las derivadas de las fuentes originales. Con ello conseguimos que la condición de pequeños retardos, en este caso particular, pueda ponerse de la forma:

$$\tau_{i,j}^2 \ll \frac{3}{2\pi^2 v_M^2} \quad \forall i, j \quad (5.27)$$

Comparando la nueva condición para los retardos con la anterior dada en (5.1) podemos deducir que, operando con las mezclas de la forma y en las condiciones descritas, pueden introducirse retardos mayores de los que en condiciones normales el algoritmo es capaz de soportar. A costa de ello, el algoritmo devolverá las derivadas de las fuentes y se tendrá que emplear un receptor más. Esto no supondrá un problema en muchas aplicaciones y siempre será posible estimar las fuentes originales a partir de las derivadas mediante integración.

5.4.2 Simulaciones realizadas y resultados obtenidos

Para comprobar la validez del método de separación propuesto, se realizaron varias simulaciones, respetando las condiciones descritas en el apartado anterior. En dichas simulaciones, se mezclaron dos fuentes con el propósito de comparar los resultados obtenidos mediante simulación con los resultados obtenidos experimentalmente por los desarrolladores del algoritmo y descritos en [1], al mezclar también dos fuentes. Téngase en cuenta que tal como se demuestra en apéndice A.2, este procedimiento es incapaz de separar mezclas de más de tres fuentes, si se utiliza en una implementación física real.

En las simulaciones se mantuvieron el resto de condiciones comentadas en el apartado 5.1.2. Las mezclas se generaron introduciendo retardos en las señales recibidas en los receptores distintos al de referencia y sumando simplemente las señales retardadas en cada receptor. Al contrario que en las simulaciones anteriores, ahora las señales retardadas no se multiplicaron por un coeficiente generado de forma aleatoria para modelar las distintas atenuaciones que sufren las fuentes en su propagación hacia los receptores, pues se asume que todas sufren la misma atenuación.

En este caso los retardos máximos admisibles vienen dados por la condición (5.27). Si en ella, imponemos que el término de la izquierda sea un orden de magnitud menor que el de la derecha, tendremos la siguiente expresión para los retardos máximos admisibles con respecto al receptor de referencia:

$$10\tau_{\max}^2 = \frac{3}{2\pi^2 v_M^2} \quad (5.28)$$

Despejando obtenemos que el retardo máximo es igual a:

$$\tau_{\max} = \frac{\sqrt{3}}{\sqrt{20\pi} f_{\max}} \quad (5.29)$$

En las condiciones del apartado 5.1.2, $f_{\max} = 2\text{KHz}$. Sustituyendo se obtiene que el retardo máximo admisible es igual a:

$$\tau_{\max} = 6.164 \cdot 10^{-5} \text{ s} \quad (5.30)$$

Las señales de audio utilizadas, al igual que antes, fueron grabadas con una tasa de muestreo de 48KHz , por lo que el número máximo de retraso en muestras que podemos tener es de:

$$6.164 \cdot 10^{-5} \text{ s} \times \frac{48000 \text{ muestras}}{\text{s}} \approx 3 \text{ muestras} \quad (5.31)$$

Comparando con los retrasos máximos admisibles que se obtuvieron en el apartado 5.1.2, vemos que en este caso, particular pueden obtenerse retardos $\sqrt{3}$ mayores.

En las diversas simulaciones realizadas, se fueron ampliando el número de retardos entre las fuentes recibidas. Además se utilizó una $SNR = 30\text{dB}$. Los resultados obtenidos fueron bastante similares a los obtenidos experimentalmente por los desarrolladores del algoritmo en [1].

A continuación se muestran la estimación de las derivadas y las derivadas de las fuentes, para una simulación en la que se introdujeron retardos de hasta tres muestras. Comparando ambas gráficas, vemos que la calidad de las estimaciones es bastante buena en las condiciones descritas.

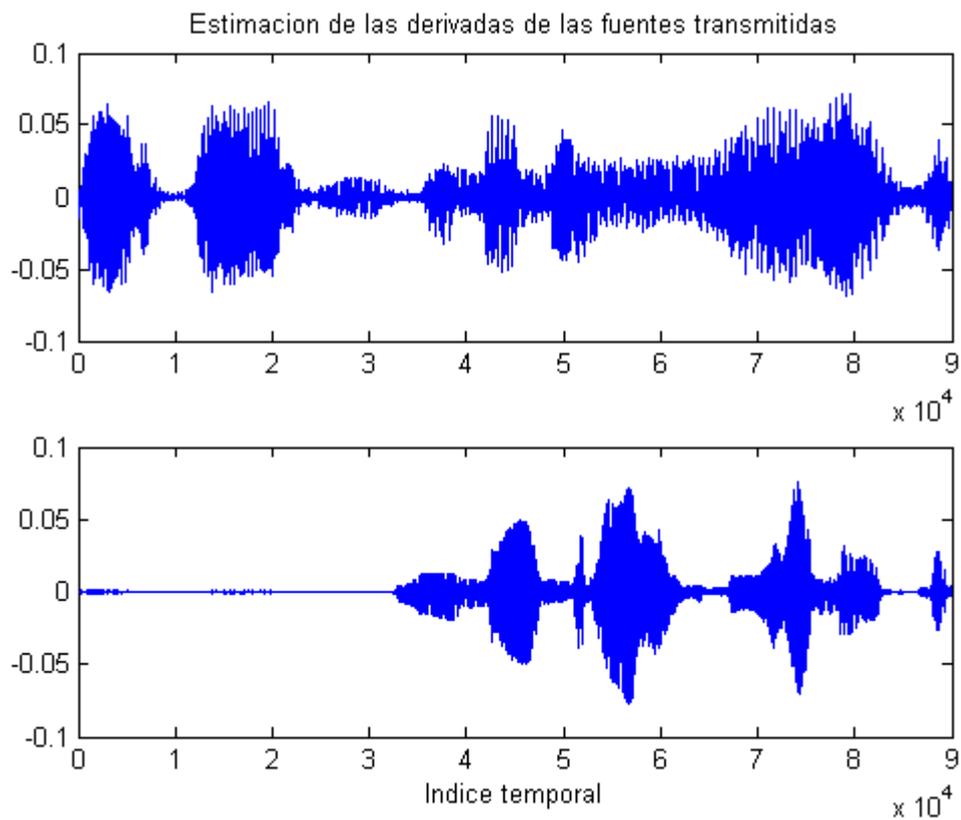


Figura 5.21: Estimación de las derivadas de las fuentes transmitidas

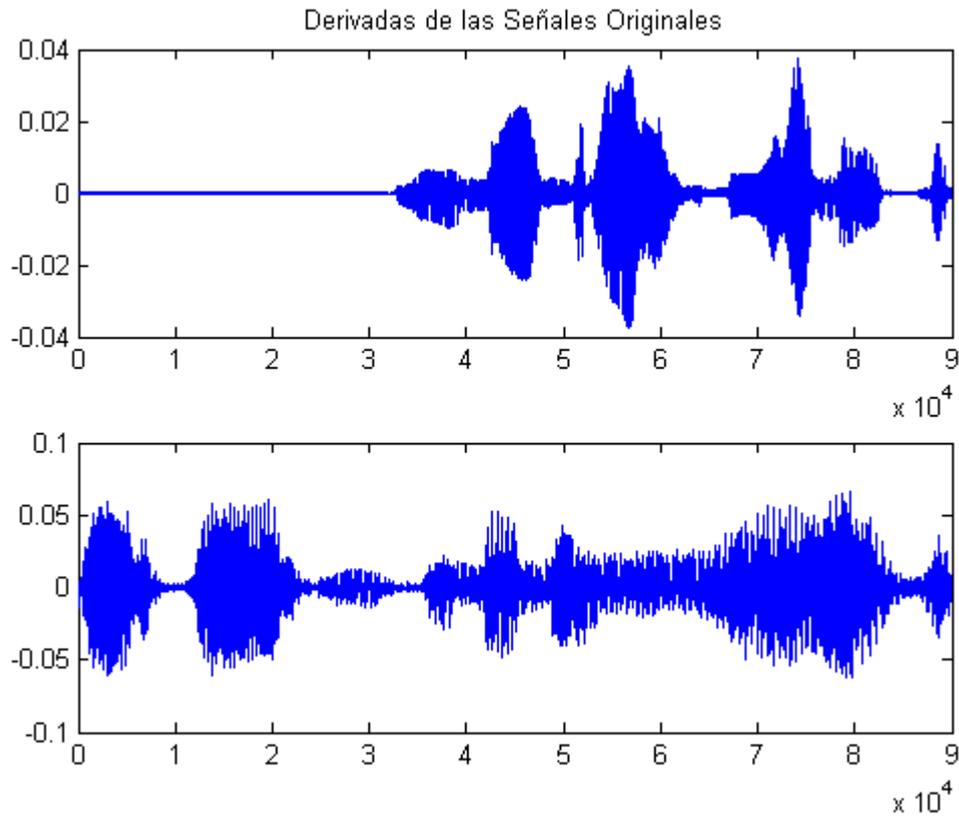


Figura 5.22: Derivadas de las fuentes transmitidas

En simulaciones sucesivas, se aumentaron poco a poco los retardos introducidos. Los resultados obtenidos fueron los siguientes:

- Para retardos de hasta 5 muestras (el límite teórico era de 3) los resultados obtenidos fueron aceptables.
- Para retardos de 6 a 8 muestras, empieza a observarse degradación en las señales estimadas. En ellas se aprecia la influencia de la otra señal que formaba parte de la mezcla.
- Para retardos por encima de las 10 muestras, el algoritmo deja de funcionar correctamente y sólo devuelve mezclas de las señales de interés.

A la hora de utilizar esta optimización del algoritmo, hemos de tener en cuenta que si se introducen retardos nulos o muy pequeños, las estimaciones no serán correctas puesto que tal como puede verse en la expresión (5.26), las señales de interés van multiplicadas por los retrasos introducidos. Si estos retrasos se hacen nulos o muy pequeños, en las observaciones dominará el término correspondiente al ruido y la separación no será posible. En las simulaciones realizadas con las condiciones descritas, a partir de retrasos de una muestra, se obtuvieron resultados aceptables.

Hemos de destacar que el comportamiento frente al ruido resultó ser mucho peor que en las versiones anteriores del algoritmo. La razón de esto es la misma de antes, las fuentes de interés están multiplicados por los retrasos que se introduce, que por definición, son pequeños. En la práctica para relaciones señal a ruido por debajo de los $25dB$, los resultados no son aceptables. Si se desea que las señales de audio no se vean muy degradadas, es conveniente trabajar con relaciones señal a ruido no inferiores a los $30dB$.

5.4.3 Conclusiones sobre el comportamiento del método de separación propuesto

A partir de los resultados obtenidos en las simulaciones, puede verse que el método propuesto puede ser una buena alternativa cuando se deseen introducir retardos algo mayores que los que soporta el método de separación general, el número de fuentes que se desean separar no es superior a tres y todas las fuentes sufran la misma atenuación en su propagación hacia los receptores.

El método tiene las limitaciones de que necesita funcionar con relaciones señal a ruido relativamente altas y de que no devuelve las señales de interés, sino sus derivadas. Aunque siempre se podrán estimar las fuentes originales integrando las estimaciones de las derivadas. En el caso de señales de voz, sus derivadas son perfectamente inteligibles, por lo que si no se desea una alta calidad en las estimaciones de las señales, podemos conformarnos con las estimaciones de las derivadas que devuelve el algoritmo.

Por último, se ha visto que en este caso la condición de pequeños retardos también resulta ser algo pesimista y puede trabajarse con retardos mayores de los previstos.

5.5 Resumen del capítulo y conclusiones

En el presente capítulo se ha estudiado el funcionamiento de una implementación en el lenguaje de programación MATLAB del algoritmo propuesto en el capítulo anterior para la separación de fuentes con retardos. Además de su correcto funcionamiento, se han comprobado la validez de las hipótesis y aproximaciones que se hicieron en el capítulo anterior. También se han recogido los resultados de numerosas simulaciones para analizar el comportamiento del algoritmo al variar los diversos parámetros que intervienen en él: número de receptores, nivel de ruido, magnitud de los retrasos, tipo de filtro empleado, etc.

Todos estos estudios se han repetido para la versión ampliada del algoritmo y, a partir de ellos, se ha hecho una comparativa entre ambas versiones en la que se han analizado las ventajas y desventajas de cada una frente a la otra.

Finalmente se ha estudiado un caso particular, en el que realizando ciertas operaciones con las observaciones, pueden obtenerse mejoras en el comportamiento frente a los retardos introducidos.

A modo de conclusión podemos decir que las hipótesis realizadas en el capítulo anterior son acertadas, puesto que los resultados obtenidos han sido los esperados. Además se ha visto cómo variando de forma adecuada ciertos parámetros, como el número de receptores, y eligiendo de forma adecuada el filtro auxiliar, el comportamiento del algoritmo mejora.

CAPÍTULO 6

CONCLUSIONES Y LÍNEAS FUTURAS

6.1 Conclusiones

En este proyecto hemos podido comprobar que la propiedad de independencia de las fuentes es suficiente para la resolución del problema de separación ciega de fuentes. Además se ha demostrado cómo la propiedad de independencia no resulta ser demasiado restrictiva y se cumple en la mayoría de los casos de interés. También hemos visto como las técnicas PCA, pese a no resolver el problema de separación debido a que el hecho de que las fuentes no estén correladas entre sí no es suficiente para poder identificarlas a partir de sus mezclas, pueden ser bastante útiles y facilitar la resolución del problema de separación ciega de fuentes. Además se ha podido comprobar, a partir de las simulaciones, las limitaciones que presentan las técnicas ICA, como la incapacidad de determinar el orden y la energía de las fuentes y el hecho de que no se puedan separar fuentes con distribuciones de tipo gaussiano.

En el proyecto se ha estudiado el comportamiento de un algoritmo ideado para la separación de fuentes que, en su propagación hacia los receptores, sufren distintos retardos. Se ha considerado que el medio por el cual se propagan las señales introduce únicamente un retardo y una atenuación en la propagación de las fuentes. Al introducirse un cierto retardo, el modelo de mezclas instantáneas estudiado en el capítulo 2 ya no será totalmente válido. Este hecho se acentúa en el caso de la separación de señales de audio o *cocktail-party problem*, pues en el caso de señales de voz, la velocidad de propagación es la del sonido y una mínima separación entre las fuentes o receptores hará que los retrasos sean apreciables. Rigurosamente para resolver el problema de separación de las fuentes, tendría que emplearse el modelo para mezclas convolutivas en ICA, lo cual dificulta mucho la separación de las fuentes a partir de las mezclas recibidas, ya que este modelo es de gran complejidad matemática. Este

problema puede simplificarse mucho, si se utiliza para la separación de las fuentes un modelo basado en mezclas instantáneas de señal. Esto último, será posible si los retardos que sufren las señales en su propagación son pequeños. Bajo estas condiciones, el problema de separación de fuentes puede abordarse como un problema de mezclas instantáneas en el que las fuentes involucradas son las de interés y sus derivadas. Este es el fundamento básico del algoritmo implementado.

La versión inicial del algoritmo propuesto tenía algunas carencias, ya que realizaba siempre la separación ciega de las fuentes de interés y de sus derivadas. Además su comportamiento frente al ruido dependía mucho del tipo de filtro auxiliar utilizado para obtener la matriz de autocovarianzas filtrada de las observaciones.

Con el fin de eliminar estas carencias, se implementó una versión mejorada del algoritmo. La dependencia con el tipo de filtro utilizado se corrigió haciendo uso de varios filtros en lugar de uno solo. La obtención de las fuentes de interés se consiguió invocando varias veces al algoritmo y realizando la extracción de sólo la fuente de mayor energía mediante el método de las potencias. Una vez extraída la fuente de mayor energía, se elimina del vector de mezclas mediante la operación de deflación y se vuelve a ejecutar el algoritmo para obtener la siguiente fuente de mayor energía presente en la mezcla.

Estas mejoras consiguieron suplir las carencias del algoritmo, pero a costa de un mayor coste computacional, que de cara a una posible implementación física del algoritmo en un futuro, no parecen justificables, a la vista de los resultados obtenidos en las diversas simulaciones, ya que la calidad de las señales obtenidas fue bastante similar en ambos casos.

6.2 Líneas futuras de investigación

A la vista de los resultados obtenidos en las simulaciones, la versión básica del algoritmo posee unas prestaciones muy similares a la de la versión mejorada y su tiempo de respuesta es mucho más bajo. Sus carencias con respecto a la versión mejorada eran la dependencia con respecto al filtro auxiliar elegido y el hecho de que no distingue entre las fuentes de interés y sus derivadas. La primera de las carencias es fácilmente subsanable, eligiendo el filtro auxiliar de forma adecuada no tendremos problemas. Para resolver la segunda de las carencias, podrían idearse algunos métodos alternativos a la deflación, combinada con el método de las potencias para el cálculo de

los autovectores, con un coste computacional menor. Para ello podríamos basarnos en la energía de las señales estimadas, ya que, en condiciones normales, la energía de las señales de interés es mucho mayor que la de las derivadas.

Una vez encontrado el algoritmo que solucione los problemas descritos, podemos pensar en una implementación física del sistema de separación que funcione de forma aproximada en tiempo real.

Durante todas las simulaciones realizadas, se ha trabajado con señales de audio. También se podría hacer un estudio del comportamiento del algoritmo, con señales de comunicaciones, en lugar de señales sonoras, ya que en cualquier sistema de comunicaciones que tenga por objeto la recepción de señales, junto con las de interés se recibirán otras y, además, las señales sufrirán un retardo en su propagación hacia los receptores, por pequeño que sea. Por lo que este tipo de situaciones son fácilmente modelables con el algoritmo propuesto.

Por último, a la vista de la gran importancia que ha demostrado tener una correcta elección de los filtros auxiliares en presencia de ruido, podría abordarse un estudio más serio, para la elección de los filtros utilizados en el algoritmo. En las simulaciones, éstos han sido elegidos sabiendo que su respuesta impulsiva debe ser nula en el origen y teniendo en cuenta las propiedades de las señales de voz. En el caso, de que se utilizasen otro tipo de señales, posiblemente los filtros escogidos no mostrarían un comportamiento óptimo.

APÉNDICE A

DEMOSTRACIONES

A.1 Dependencias entre un proceso estacionario de primer orden y su derivada primera

Sea $\mathbf{x}(t)$ un proceso estacionario hasta segundo orden, los procesos $\mathbf{x}(t)$ y $\dot{\mathbf{x}}(t)$ no están correlados para un instante de tiempo dado.

Demostración:

Al ser $\mathbf{x}(t)$ un proceso estacionario hasta segundo orden, se cumple:

$$E[\mathbf{x}] = \eta_x \text{ y } Var[\mathbf{x}] = \sigma_x^2 \quad (\text{A.1})$$

Y además tendremos que:

$$E[\dot{\mathbf{x}}] = \frac{d}{dx}(E[\mathbf{x}]) = 0 \quad (\text{A.2})$$

Para que los procesos no estén correlados, su matriz de autocovarianzas debe ser nula:

$$\text{cov}(\mathbf{x}, \dot{\mathbf{x}}) = E[\mathbf{x}\dot{\mathbf{x}}^T] - E[\mathbf{x}]E[\dot{\mathbf{x}}] = E[\mathbf{x}\dot{\mathbf{x}}^T] = 0 \quad (\text{A.3})$$

Como veremos a continuación, se cumple que:

$$\mathbf{R}_{\mathbf{x}\mathbf{x}}(0) = E[\mathbf{x}\mathbf{x}^T] = 0 \quad (\text{A.4})$$

Para un sistema con respuesta impulsiva $h(t)$ se cumple que:

$$\mathbf{S}_{\mathbf{xy}}(w) = \mathbf{S}_{\mathbf{xx}}(w) \cdot \mathbf{H}^*(w) \quad (\text{A.5})$$

Siendo $\mathbf{y}(t)$ y $\mathbf{x}(t)$ las salidas y entradas del sistema respectivamente. Por tanto, para un derivador tendremos:

$$\mathbf{S}_{\mathbf{xx}}(w) = -jw\mathbf{S}_{\mathbf{xx}}(w) \quad (\text{A.6})$$

Pasando al dominio temporal, se obtiene la siguiente relación:

$$\mathbf{R}_{\mathbf{xx}}(\tau) = \frac{d\mathbf{R}_{\mathbf{xx}}(\tau)}{d\tau} \quad (\text{A.7})$$

Por otra parte $\mathbf{x}(t) = \mathbf{A}\mathbf{s}(t)$, por tanto, podemos escribir:

$$\mathbf{R}_{\mathbf{xx}}(\tau) = \frac{dE[\mathbf{A}\mathbf{s}(t)\mathbf{s}^H(t-\tau)\mathbf{A}^H]}{d\tau} = \mathbf{A} \frac{dE[\mathbf{s}(t)\mathbf{s}^H(t-\tau)]}{d\tau} \mathbf{A}^H \quad (\text{A.8})$$

La matriz $E[\mathbf{s}(t)\mathbf{s}^H(t-\tau)]$ es diagonal, ya que las fuentes son estadísticamente independientes entre sí. Los elementos de la diagonal vendrán dados por:

$$\left. \frac{dR_{s_i s_i}(\tau)}{d\tau} \right|_{\tau=0} = 0 \quad \text{con } i = 1, \dots, N \quad (\text{A.9})$$

Además $R_{s_i s_i}(\tau) = E[s_i(t)s_i^H(t+\tau)]$ es, por definición, la función autocorrelación de la fuente i -ésima.

Para $\tau = 0$ se cumple que $\mathbf{R}_{\mathbf{x}\mathbf{x}}(0) = 0$ porque los elementos de la diagonal de la matriz $E[\mathbf{s}(t)\mathbf{s}^H(t-\tau)]$ son nulos, puesto que la función autocorrelación presenta un máximo en $\tau = 0$ y, por tanto, su derivada en ese punto se anula.

A.2 Limitaciones del método de separación propuesto para el caso de un medio que no introduce atenuación

Si las diferencias entre las distancias que separan las fuentes de los receptores son pequeñas, podemos considerar que todas las fuentes son recibidas en los receptores con la misma potencia, tal como se expuso en el capítulo 4.

Si en todos los receptores se reciben las señales con la misma potencia, el método de separación propuesto puede hacerse más robusto frente a los retardos si a las observaciones de los receptores distintos al de referencia les restamos la del de referencia. En este caso se obtendrá una nueva matriz de mezclas $\mathbf{A} = [\mathbf{A}_3 \quad \mathbf{A}_4]$ de dimensiones $(M-1) \times 2N$.

Para que el algoritmo de separación funcione, todas las filas de la matriz \mathbf{A}_3 debían ser linealmente independientes. Tal como se vio, la matriz \mathbf{A}_3 es de la forma:

$$\mathbf{A}_3 = - \begin{bmatrix} \tau_{2,1} & \cdots & \tau_{2,j} & \cdots & \tau_{2,N} \\ \vdots & & \vdots & & \vdots \\ \tau_{M,1} & \cdots & \tau_{M,j} & \cdots & \tau_{M,N} \end{bmatrix} \quad (\text{A.10})$$

Si los receptores están suficientemente alejados de las fuentes, podemos considerar que las fuentes se propagan en forma de frentes de onda, caracterizados por un vector de onda \mathbf{v}_j normal al plano e independiente de la posición de las fuentes. Los retardos relativos τ_{ij} sólo dependen de la distancia entre el sensor de referencia y el i -ésimo sensor, tal como se ilustra en la *figura A.1*. El vector \mathbf{u}_i indica la posición relativa del sensor i -ésimo con respecto al de referencia.

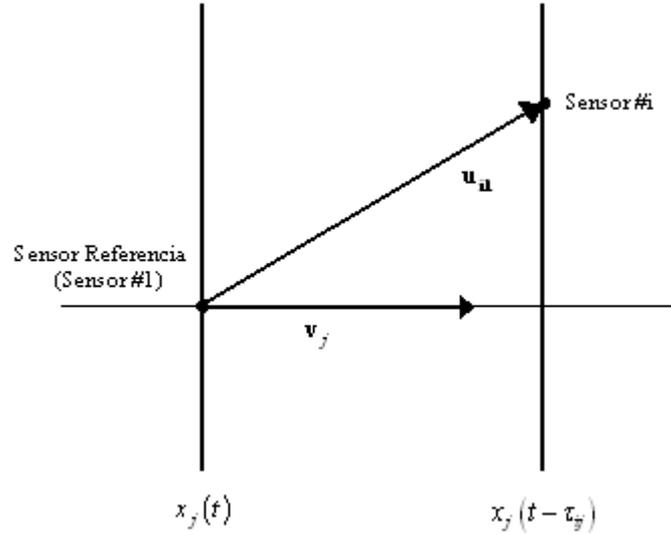


Figura A.1: Representación de los frentes de onda y de los receptores

El retardo relativo vendrá dado por el producto escalar,

$$\tau_{ij} = \frac{1}{c} \mathbf{u}_{li}^T \mathbf{v}_j \quad (4.11)$$

Donde c es la velocidad de propagación en el medio. Como la dirección de propagación de los frentes de onda no depende de la posición de las fuentes, en el espacio tridimensional un conjunto de más de tres fuentes es linealmente dependiente:

$$\mathbf{v}_4 = \sum_{j=1}^3 \alpha_j \mathbf{v}_j \quad (A.12)$$

De aquí puede deducirse que en el espacio tridimensional, la matriz

$$\mathbf{A}_3 = \frac{1}{c} [\mathbf{u}_{li}^T \mathbf{v}_j], \quad i = 2, \dots, M; \quad j = 1, \dots, N \quad (A.13)$$

no tendrá más de tres filas linealmente independientes. En otras palabras, la identificación de más de tres fuentes no será posible.

APÉNDICE B

RESOLUCIÓN DE PROBLEMAS ADICIONALES

B.1 Cálculo de los autovectores de una matriz

Para el cálculo de los autovectores de la matriz de autocovarianzas filtrada, se decidió utilizar el método de las potencias. El método de las potencias se aplica a una matriz \mathbf{A} de dimensiones $n \times n$ con un autovalor estrictamente dominante λ_1 , lo que significa que λ_1 debe ser mayor en valor absoluto que cualquier otro autovalor. En este caso, el método de las potencias produce una sucesión escalar que se aproxima a λ_1 y una sucesión vectorial que se aproxima al autovector correspondiente.

Por sencillez, supondremos que \mathbf{A} es diagonalizable y que \mathbb{R}^n tiene una base de autovectores $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_n$, de manera que sus valores propios correspondientes $\lambda_1, \lambda_2, \dots, \lambda_n$ disminuyan de tamaño con el valor propio estrictamente dominante en primer lugar. Esto es,

$$|\lambda_1| > |\lambda_2| \geq |\lambda_3| \geq \dots \geq |\lambda_n| \quad (\text{B.1})$$

Cualquier vector \mathbf{x} perteneciente a \mathbb{R}^n puede expresarse como combinación lineal de los autovectores de la matriz \mathbf{A} , por tanto, podremos escribir $\mathbf{x} = c_1 \mathbf{v}_1 + \dots + c_n \mathbf{v}_n$. Entonces se cumplirá para cualquier entero $k = 1, 2, \dots$ que:

$$\mathbf{A}^k \mathbf{x} = c_1 (\lambda_1)^k \mathbf{v}_1 + \dots + c_n (\lambda_n)^k \mathbf{v}_n \quad (\text{B.2})$$

Si $c_1 \neq 0$. Entonces, dividiendo entre $(\lambda_1)^k$;

$$\frac{1}{(\lambda_1)^k} \mathbf{A}^k \mathbf{x} = c_1 \mathbf{v}_1 + c_2 \left(\frac{\lambda_2}{\lambda_1} \right)^k + \dots + c_n \left(\frac{\lambda_n}{\lambda_1} \right)^k \mathbf{v}_n \quad (\text{B.3})$$

Todas las fracciones $\lambda_2/\lambda_1, \dots, \lambda_n/\lambda_1$ son de módulo menor que 1, de manera que cuando $k \rightarrow \infty$, sus potencias tienden a cero. Por tanto:

$$(\lambda_1)^{-k} \mathbf{A}^k \mathbf{x} \rightarrow c_1 \mathbf{v}_1 \quad (\text{B.4})$$

Entonces para k grande el método nos devuelve el vector $c_1 \mathbf{v}_1$ que tiene la misma dirección que el autovector \mathbf{v}_1 . Realmente lo único que nos interesa en el algoritmo de separación propuesto es la dirección del autovector, puesto que el algoritmo es incapaz de determinar las energías de las señales. Para simplificar el cálculo, podemos normalizar los valores de $c_1 \mathbf{v}_1$ que se van obteniendo en cada una de las iteraciones. A modo de resumen, los pasos a seguir para estimar el autovector asociado al autovalor dominante de una matriz son los siguientes:

1. Seleccionamos un vector inicial \mathbf{x}_0 normalizado
2. Para $k = 1, 2, \dots$,
 - a. Calculamos $\mathbf{x}_k = \mathbf{A} \mathbf{x}_{k-1}$.
 - b. Normalizamos \mathbf{x}_k .
3. Para casi todas las elecciones de \mathbf{x}_0 , la sucesión $\{\mathbf{x}_k\}$ se aproxima al valor deseado

BIBLIOGRAFÍA

- [1] J. Barrère, G. Chabriel, “*A Compact Sensor Array for Blind Separation of Sources*” IEEE Transactions on Circuits and Systems-I: Fundamental Theory and Applications, Vol 49, NO. 5.
- [2] A. Blanc-Lapierre , R. Fortet, “*Théorie des fonctions aléatoires,*”, Masson et Cie, Paris, France,1953
- [3] G. Chabriel, J. Barreré, “*Blind Identification of Slightly Delayed Mixtures*”, IEEE Transactions on Circuits and Systems
- [4] S. Cruces, “*Una visión unificada de los algoritmos de separación ciega de fuentes*”, Tesis Doctoral, Universidad de Vigo (1999)
- [5] D.M., Etter, “*Solución de problemas de ingeniería con MATLAB*”, Prentice-Hall, 1998
- [6] D. Hanselman, B. Littlefield, “*The Student Edition of MATLAB*”, Prentice-Hall, 1997
- [7] E. Hernández, “*Álgebra y geometría*”, Addison-Wesley,1994
- [8] A. Hyvärinen, J. Karhunen, E. Oja. “*Independent Component Análisis*”, Wiley_Interscience, 2001
- [9] D. C. Lay, “*Álgebra Lineal y sus Aplicaciones*”, Addison-Wesley,1999

BIBLIOGRAFÍA

- [10] S. Nakamura, “*Análisis numérico y visualización gráfica con MATLAB*”. Prentice-Hall, 1997

- [11] B. Noble y J. W. Daniel, “*Álgebra Lineal Aplicada2*”, Ed. Prentice-Hall, 1988

- [12] A. Papoulis, “*Probability, Random Variables, and Stochastic Process*”, New York: McGraw-Hill, 1984

- [13] Philippe G. Ciarlet, “*Introduction to numerical and linear algebra and optimisation*”, Cambridge University Press, 1989

- [14] L. Tong, R.-W. Liu, V.C. Soon, and Y.-F. Huang. “*Indeterminacy and identifiability of blind identification*”, *IEEE Trans. On Circuits and Systems*, 38:499-509, 1991