



UNIVERSIDAD DE GRANADA
Facultad de Ciencias
Fecha ... 3/9/99 ...
ENTRADA NUM. ... 2939 ...

UNIVERSIDAD DE GRANADA
27 JUL. 1999
COMISION DE DOCTORADO

**NUEVOS ALGORITMOS PARA SEPARACIÓN
CIEGA DE FUENTES UTILIZANDO
MÉTODOS GEOMÉTRICOS**

Beatriz Prieto Campos

TESIS DOCTORAL

Directores:

Alberto Prieto Espinosa
Carlos García Puntonet

1999

DEPARTAMENTO DE ARQUITECTURA Y TECNOLOGÍA DE COMPUTADORES

7 6 21



**NUEVOS ALGORITMOS PARA SEPARACIÓN
CIEGA DE FUENTES UTILIZANDO
MÉTODOS GEOMÉTRICOS**

BIBLIOTECA UNIVERSITARIA
GRANADA
N.º Documento 613383772
N.º Copia 15587125

Beatriz Prieto Campos

TESIS DOCTORAL

Directores:

Alberto Prieto Espinosa
Carlos García Puntonet

1999

DEPARTAMENTO DE ARQUITECTURA Y TECNOLOGÍA DE COMPUTADORES

D. Alberto Prieto Espinosa, Catedrático de Universidad y D. Carlos García Puntonet, Profesor Titular de Universidad, del Departamento de Arquitectura y Tecnología de Computadores

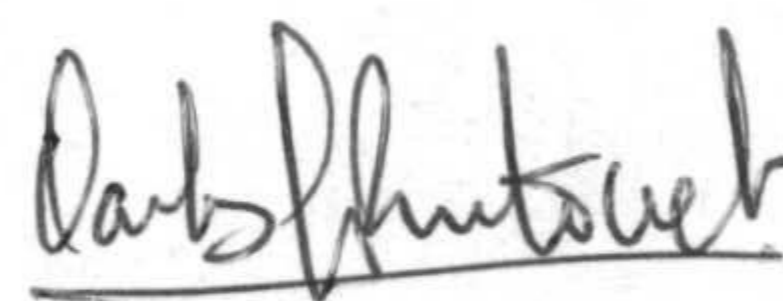
CERTIFICAN

Que la memoria titulada: "*Nuevos algoritmos para separación ciega de fuentes utilizando métodos geométricos*", ha sido realizada por **Dña. Beatriz Prieto Campos** bajo nuestra dirección en el Departamento de Arquitectura y Tecnología de Computadores de la Universidad de Granada para optar al grado de Doctora en Ingeniería Electrónica.

Granada, a 26 de Julio de 1999



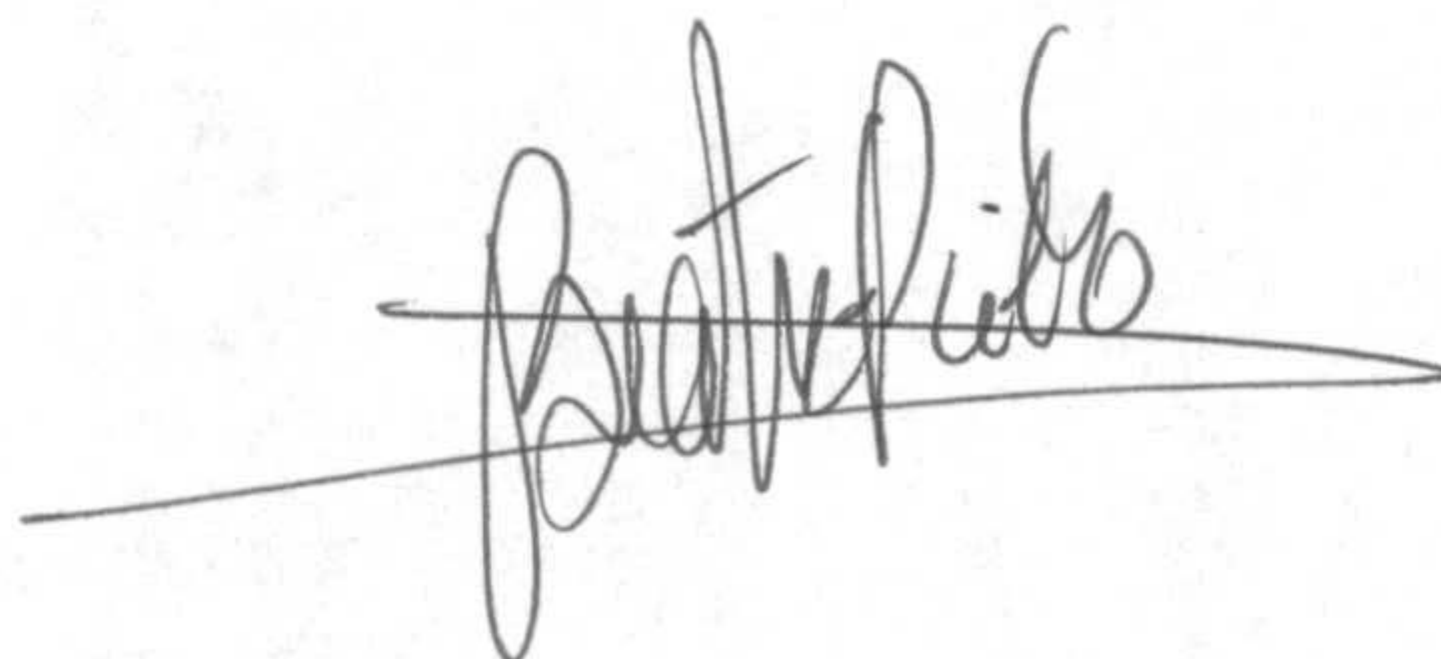
Fdo. Alberto Prieto Espinosa
Director de Tesis



Fdo. Carlos García Puntonet
Director de Tesis

**NUEVOS ALGORITMOS PARA SEPARACIÓN
CIEGA DE FUENTES UTILIZANDO
MÉTODOS GEOMÉTRICOS**

Beatriz Prieto Campos

A handwritten signature in black ink, appearing to read 'Beatriz Prieto Campos', with a long horizontal line extending to the right from the end of the signature.

Fdo. Beatriz Prieto Campos

A todos los que me han ayudado

AGRADECIMIENTOS

A mis directores de tesis, D. Alberto Prieto Espinosa y D. Carlos G. Puntonet, por su apoyo e inestimable ayuda durante los años de realización de esta memoria. Por sus ideas y orientación sin las cuales este proyecto no se habría llevado a cabo. También les agradezco sus correcciones al manuscrito.

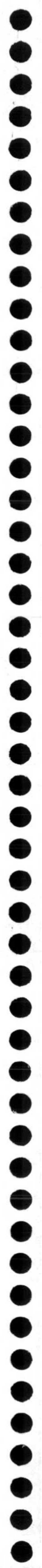
A mis compañeros de departamento en especial a David Palomar, Antonio Cañas, Antonio Díaz y Begoña del Pino, por su colaboración y por responder pacientemente a mis continuas preguntas.

A Christian Jutten por haberme acogido en su Laboratoire des Images et des Signaux en el INPG de Grenoble, y por sus acertadas sugerencias.

A Te-Won Lee y Terry Sejnowski por darme nuevas visiones e ideas en mi estancia en el Computational Neurobiological Laboratory (CNL) del Salk Institute de San Diego y seguir colaborando vía correo electrónico.

A Juan de Dios García López-Durán del Departamento de Física Aplicada le agradezco su asesoramiento en los aspectos relacionados con estadística.

A todos los que han estado próximos a mí durante la realización de esta memoria.



PRÓLOGO

El problema de separación ciega de fuentes fue enunciado por primera vez en 1985 por J. Héroult, C. Jutten y B. Ans [HER85], y consiste en la obtención de las señales generadas por p fuentes a partir de las mezclas detectadas por q sensores, conociendo tan sólo estas últimas. La mezcla de señales tiene lugar en el medio en que se propagan y en los sensores. El concepto de la separación ciega de fuentes ha abierto un capítulo de gran relevancia en la Teoría de Señales, con aplicaciones en una gran diversidad de campos (radiocomunicaciones, procesamiento de voz y de imágenes, biomedicina, sensores inteligentes, etc.).

Los trabajos presentados hasta ahora para resolver el problema de la separación ciega de fuentes se pueden encuadrar dentro de uno de dos enfoques: estadístico o geométrico.

El *enfoque estadístico* parte de la hipótesis de que las señales originales (fuentes) son estadísticamente independientes y los procedimientos que lo siguen se basan en propiedades de las fuentes y de las observaciones (mezclas), caracterizadas por sus distribuciones de probabilidad o por sus cumulantes. Se pretende que las señales recuperadas (estimadores de las fuentes) sean tales que cumplan un determinado criterio basado en principios estadísticos o de la Teoría de la Información. El criterio se suele modelar matemáticamente con una *función de contraste*, cuya minimización conduce a la estimación de la matriz que modela el proceso de mezcla (*matriz de mezcla*). En definitiva, se considera que si las señales originales son estadísticamente independientes, las señales recuperadas también deben de serlo.

El *enfoque geométrico* fue introducido en 1995 por C.G.Puntonet y colaboradores [PUN95] y trata de explotar las propiedades geométricas y algebraicas de las señales originales y de las observaciones. Los fundamentos de los procedimientos que siguen este enfoque son muy sencillos y directos de aplicar, dando lugar a algoritmos de baja complejidad. No obstante, no han sido estudiados y formalizados de forma completa, la mayor parte de los resultados y conclusiones sobre su convergencia, sólo se han obtenido empíricamente, y o no son equivariantes (las prestaciones dependen de la mezcla) o su equivarianza no ha sido probada.

El presente trabajo¹ pretende contribuir al desarrollo de algoritmos de separación de señales siguiendo el enfoque geométrico, y se ha realizado tratando de cubrir los siguientes objetivos:

- (A) Presentar de forma completa y compacta los fundamentos teóricos de los métodos de separación ciega de fuentes basados en propiedades geométricas (Capítulo 1).
- (B) Concebir e implementar un entorno de simulación para desarrollar y probar algoritmos de separación ciega de señales (Capítulo 3).
- (C) Desarrollar nuevos algoritmos geométricos de separación con prestaciones superiores a los presentados hasta ahora (Capítulos 4 y 5).

¹ Parcialmente financiado por el proyecto CICYT: "Algoritmos y Arquitecturas para Separación de Señales (ALYASS)". TIC98-0982.

La presente memoria, que trata de describir el trabajo realizado, se ha estructurado en los siguientes capítulos y apéndices:

- **Capítulo 1.** Se realiza una introducción general al problema de la separación ciega de fuentes. A continuación, después de hacer una abstracción de los principales conceptos que han inspirado los trabajos más relevantes publicados en el campo, se presentan aportaciones concretas, tanto en algoritmos como en arquitecturas, realizadas bajo el enfoque estadístico y considerando un modelo de mezcla lineal e instantáneo. También se enumeran algunas de las aplicaciones de separación ciega de fuentes que han despertado mayor interés, y se describen someramente las líneas de investigación abiertas en la actualidad dentro del campo, incluyendo las aportaciones realizadas en modelos de mezcla más complejos que el lineal instantáneo.
- **Capítulo 2.** Se aborda el problema de la separación ciega de señales haciendo un estudio teórico del mismo desde un enfoque geométrico. Se razonan y justifican las hipótesis en que se basan los procedimientos geométricos y se demuestran las propiedades algebraicas y geométricas de la transformación que se produce entre las fuentes y las mezclas. Se incluyen también distintos índices o métricas para determinar las prestaciones de los resultados obtenidos con los algoritmos de separación.
- **Capítulo 3.** Se presenta un entorno software proyectado para el desarrollo y test de algoritmos de separación de mezclas lineales. El entorno se ha implementado en MATLAB, es de una gran modularidad y permite visualizar entre cualquiera de las fases de simulación las diversas señales involucradas en el proceso en distintos formatos (en el dominio del tiempo, frecuencias de amplitudes, espacios vectoriales, etc.). Este entorno se utiliza para desarrollar y probar los algoritmos de recuperación de señales que se describen en los Capítulos 4 y 5.
- **Capítulo 4.** Se presenta un algoritmo que trata de estimar un vértice del hiperparalelepípedo que forman los vectores de mezcla y un vector en cada una de las p aristas incidentes en dicho vértice. Este conjunto de puntos, tomando como origen el vértice, constituyen p vectores que considerados como columnas forman una matriz que, según se demuestra en el Capítulo 2, estima la matriz de mezcla. El algoritmo se implementa con una red neuronal que, una vez obtenido el vértice, selecciona los p vectores de mezcla con mayor separación angular entre ellos, los cuales coinciden con los vectores buscados ya que pertenecerán a las aristas. Los pesos se obtienen mediante

un aprendizaje competitivo, y se identifican con los elementos de la matriz de estimación de la mezcla. La recuperación de señales se efectúa con otra red neuronal, de tipo recursivo, y con pesos comunes a la red de aprendizaje.

- **Capítulo 5.** El algoritmo descrito en el Capítulo 4 es poco eficiente cuando se trata de separar señales en las que la probabilidad de generación de sus valores extremos (máximos y mínimos) es baja; es decir, señales cuya función de densidad de probabilidad es tal que sus valores en los límites de su rango son menores que la media. En este Capítulo se presenta otro algoritmo original aplicable a las señales citadas; es decir, con funciones de densidad de probabilidad unimodales y centradas (tales como señales de voz y música). El algoritmo se basa en el hecho de que los ejes de la distribución de vectores de mezcla son paralelos a las aristas del hiperparalelepípedo que contiene los puntos de la mezcla. El algoritmo determina las pendientes de dichos ejes, que coinciden con las de las aristas, y con ellos, según se demuestra en el Capítulo 2, se puede formar una matriz que estima a la matriz de mezcla.
- **Capítulo 6.** Recoge las conclusiones y las principales aportaciones del presente trabajo, así como las líneas de investigación que abre, de interés para desarrollar en un futuro.
- **Apéndice 1.** Incluye algunos conceptos básicos de estadística que ayudan a comprender mejor los contenidos de la presente memoria, ya que con ellos se tiene una referencia próxima y en la notación que se usa a lo largo de los distintos capítulos.
- **Apéndice 2.** Agrupa una serie de conceptos y propiedades de Teoría Estadística de la Información en los que se basan algunos de los métodos más renombrados de separación ciega de fuentes, y que se describen en el Capítulo 1.
- **Apéndice 3.** Contiene los resultados de simulación obtenidos con el algoritmo de determinación de los ejes de la distribución de vectores de mezcla sirviendo, por tanto, de apoyo y complemento al Capítulo 5.
- **Apéndice 4.** Describe la interrelación entre los distintos programas en MATLAB que conforman el entorno de simulación descrito en el Capítulo 3.

NOTACIÓN

Las referencias a secciones (Sec.c.n), figuras (Figura c.n), tablas (Tabla c.n), expresiones ((c.n)), hipótesis (Hipótesis c.n), teoremas (Teorema c.n), y propiedades (Propiedad c.n), se hacen incluyendo en primer lugar el número, *c*, de capítulo (1,2,... ó 6) o apéndice (A1, A2 ó A3) del que se trata, seguido de un número, *n*, de orden correlativo dentro del capítulo o apéndice.

Las referencias bibliográficas se codifican entre corchetes, con las tres primeras letras del primer apellido del primer autor, seguidas de las dos últimas cifras del año de publicación. Caso de que en el mismo año el primer autor tenga varias referencias, se incluye una letra minúscula (a, b, c,...) adicional, para identificar cada una de ellas. Las referencias bibliográficas se incluyen al final de la memoria ordenadas según el código indicado anteriormente: orden alfabético / año de publicación.

ACRÓNIMOS Y SÍMBOLOS UTILIZADOS EN LA PRESENTE MEMORIA

ANN	Red neuronal artificial (“Artificial Neural Network”)
GHA	Algoritmo hebbiano generalizado (“Generalized Hebbian Algorithm”)
ICA	Análisis de Componentes Independientes (“Independent Component Analysis”)
ME	Máxima Entropía (“Maximum Entropy”)
MI	Información Mutua (“Mutual Information”)
ML	Máxima verosimilitud (“Maximum Likelihood”)
PCA	Análisis de Componentes Principales (“Principal Components Analysis”)
RMS	Error Cuadrático Medio (“Root Mean Square”)
SNR	Relación Señal/Ruido (“Signal to Noise Rate”)
σ	Desviación estándar
Φ	Función de contraste
μ	Valor medio
$N(\mu, \sigma^2)$	Distribución gaussiana de media μ y varianza σ^2
$\eta(m)$	Ganancia de aprendizaje en el instante m
Φ^0	Función de contraste ortogonal
σ^2	Varianza
λ_j	Valores propios
$\kappa_{y_i, k}$	Cumulante (auto-cumulante) de orden k de la variable aleatoria y_i
$\langle s_i \rangle$	valor medio de la variable aleatoria s_i
A	Matriz de mezcla
A_d	Matriz canónica de A (diagonal principal a unos)
B	Matriz de reconstrucción de las fuentes a partir de las observaciones
$C_{1\dots k}[y]=\text{Cum}[y_1, \dots, y_k]=\kappa_{y, 1\dots k}$	Cumulante de orden k de las variables aleatorias y_1, \dots, y_k
D	Matriz diagonal
$D(f_1 f_2; \mathbf{B})$	Divergencia de Kullback-Leibler entre las funciones de distribución f_1 y f_2 expresada en función de B
$E\{x\}$	Esperanza matemática de la variable x

$e=(e_i)^T$	Vector de señales observadas
<i>exponente -1</i>	Inversa de una matriz (\mathbf{A}^{-1} es la matriz inversa de \mathbf{A})
<i>exponente T</i>	Traspuesta (\mathbf{A}^T es la matriz traspuesta de \mathbf{A} ; \mathbf{s}^T es el vector traspuesto de \mathbf{s})
F_s	Frecuencia de muestreo
$f_{s_i}(s_{s_i})$	Función de densidad de probabilidad de la señal aleatoria s_i
\mathbf{I}	Matriz diagonal unitaria
$L(\bullet)$	Función logarítmica de máxima verosimilitud
m	Número de muestra, paso de iteración $t=mT_s$
M	Número de muestras
$\mathbf{n}=(n_i(t))^T$	Vector de ruido
n_c	Número de clases
\mathbf{P}	Matriz de prestaciones
p	Número de fuentes
\mathbf{R}	Matriz de correlación
$\mathbf{s}=(s_i)^T$	Vector de fuentes o señales originales
subíndice M	Índice de valor máximo
subíndice m	Índice de valor mínimo
T_s	Periodo de muestreo
$U(s_m, s_M)$	Distribución uniforme de rango $[s_m, s_M]$
\mathbf{V}	Matriz DE transformación de blanqueo
\mathbf{W}	Matriz de pesos de una red neuronal, o matriz que estima a la matriz de mezcla \mathbf{A} , dependiendo del contexto.
$\mathbf{y}=(y_i)^T$	Vector de señales recuperadas o estimadas
\mathbf{Z}	Matriz de permutación

ÍNDICE

PRÓLOGO.....	xi
ACRÓNIMOS Y SÍMBOLOS UTILIZADOS EN LA PRESENTE MEMORIA.....	xvii
CAPÍTULO 1: REVISIÓN DE LOS MÉTODOS DE SEPARACIÓN CIEGA DE FUENTES	1
1.1 PLANTEAMIENTO/DESCRIPCIÓN DEL PROBLEMA	2
1.2 INDETERMINACIONES E HIPÓTESIS	7
1.3 PRINCIPIOS DE SEPARACIÓN.....	9
1.3.1. Criterios estadísticos de separación. Funciones de contraste.....	11
1.3.2. Análisis de Componentes Independientes (ICA)	14
1.4 ARQUITECTURAS PARA SEPARACIÓN DE FUENTES.....	14
1.5 BLANQUEADO ESPACIAL. ANÁLISIS DE COMPONENTES PRINCIPALES.....	19
1.6 EL PROCEDIMIENTO PIONERO DE SEPARACIÓN DE FUENTES: ARQUITECTURA NEUROMIMÉTICA DE JUTTEN-HERAULT.....	23
1.7 MÉTODOS BASADOS EN LA MINIMIZACIÓN DE LA DEPENDENCIA ESTADÍSTICA (ICA)	26
1.8 ESTIMACIÓN DE MÁXIMA VEROSIMILITUD (ML).....	31
1.9 MÉTODOS BASADOS EN EL PRINCIPIO DE MÁXIMA ENTROPÍA.....	34
1.10 MÉTODOS BASADOS EN MEDIDAS ESTADÍSTICAS DE ALTO ORDEN.....	40
1.11 ANÁLISIS DE COMPONENTES PRINCIPALES NO LINEAL.....	43
1.12 OTROS ALGORITMOS.....	46
1.12.1 Red neuronal recursiva: Cichocki	46
1.12.2 Otras redes propuestas por Cichocki y Amari.....	51
1.13 APLICACIONES DE LA SEPARACIÓN CIEGA DE FUENTES	52
1.14 AMPLIACIONES DEL MODELO BÁSICO. LÍNEAS ABIERTAS	55
1.14.1 Datos y mezclas no estacionarios.....	56
1.14.2 Efecto del ruido.....	56
1.14.3 Estimación del número de fuentes, p	57
1.14.4 Mezclas mal condicionadas	58
1.14.5 Modelos de mezcla más complejos que el lineal instantáneo.	58
1.14.6 Uso de información a priori.	58
1.15 CONCLUSIONES	59
CAPÍTULO 2: FUNDAMENTO TEÓRICO DE LOS MÉTODOS GEOMÉTRICOS DE SEPARACIÓN DE FUENTES.....	61
2.1. MODELO MATEMÁTICO DE LA MEZCLA LINEAL DE SEÑALES.	62
2.1.1 Hipótesis del procedimiento.....	63
2.1.2 Modelo con ruido aditivo.....	67
2.2 PROPIEDADES ANALÍTICAS Y GEOMÉTRICAS DE LA TRANSFORMACIÓN LINEAL DE LAS FUENTES	69
2.3 FUNDAMENTO DEL MÉTODO GEOMÉTRICO DE SEPARACIÓN DE FUENTES.....	78
2.4 OBTENCIÓN DE LOS VÉRTICES DEL HIPERPARALELEPÍPEDO	

	DEL ESPACIO DE OBSERVACIONES	85
	2.4.1 Relación entre los vértices	90
	2.4.2 Ejemplo	94
2.5	MEDIDAS DE LAS PRESTACIONES	96
	2.5.1 Índice de prestaciones de Amari	96
	2.5.2 Error cuadrático medio.....	98
	2.5.2.1 <i>Error cuadrático medio entre los elementos de W y A</i>	99
	2.5.2.2. <i>Error cuadrático medio entre las fuentes originales y las señales</i> <i>reconstruidas.</i>	102
	2.5.3 Diafonía.....	105
2.6	CONCLUSIONES	106

CAPÍTULO 3: ENTORNO DE SIMULACIÓN PARA DESARROLLAR ALGORITMOS DE SEPARACIÓN DE MEZCLAS LINEALES 109

3.1	INTRODUCCIÓN	110
3.2	ESTRUCTURA GENERAL DEL SIMULADOR	111
3.3	MÓDULO DE GENERACIÓN DE SEÑALES	112
	3.3.1 Señales sintéticas disponibles en el simulador.....	112
	3.3.2 Señales reales incluidas en el simulador	121
3.4	MÓDULO PARA SIMULAR LA MEZCLA.....	122
3.5	MÓDULO PARA LA ESTIMACIÓN DE LA MATRIZ DE MEZCLA.....	123
3.6	RECONSTRUCCIÓN DE LAS FUENTES	125
3.7	MÓDULO DE MEDIDAS DE LAS PRESTACIONES	127
3.8	MÓDULO DE RESULTADOS Y REPRESENTACIONES	129
3.9	CONCLUSIONES	133

CAPÍTULO 4: ALGORITMO DE MÁXIMA SEPARACIÓN ANGULAR PARA SEPARACIÓN CIEGA DE FUENTES..... 135

4.1.	PRINCIPIOS DEL MÉTODO	136
4.2	TRASLACIÓN DEL HIPERPARALELPÍPEDO	140
4.3	SEÑALES CON RUIDO	142
4.4	VECTORES CRÍTICOS.....	145
	4.4.1 Caso de tres fuentes, $p=3$	147
4.5	IMPLEMENTACIÓN DEL ALGORITMO CON REDES NEURONALES	147
	4.5.1 Red de separación de las señales	148
	4.5.1.1 <i>Caso $p=3$</i>	
	4.5.2 Red de aprendizaje.....	150
	4.5.2.1 <i>Caso $p=3$</i>	
4.6	IMPLEMENTACIÓN.....	155
4.7	RESULTADOS EXPERIMENTALES	157
4.8	CONCLUSIONES	165

CAPÍTULO 5: ALGORITMO DE LOCALIZACIÓN DE LOS EJES DE LA DISTRIBUCIÓN DE VECTORES DE MEZCLA PARA SEPARACIÓN CIEGA DE SEÑALES..... 167

5.1	SEPARACIÓN DE MEZCLAS DE SEÑALES DE VOZ	169
5.2	BASES DEL PROCEDIMIENTO	171
5.3	ALGORITMOS PARA SEPARACIÓN DE SEÑALES UNIMODALES CENTRADAS	173
	5.3.1 Algoritmo adaptativo en línea	174
	5.3.2 Implementación del algoritmo	177
	5.3.3 Algoritmo fuera de línea.....	181
5.4	RESULTADOS EXPERIMENTALES.	183
	5.4.1 Resultados con señales sintéticas.....	185
	190	
5.5	DEPENDENCIA DE LA ESTIMACIÓN DE LOS VALORES DE LA MATRIZ DE MEZCLA CON EL NÚMERO DE CLASES CONSIDERADAS.....	197
5.6	DISCUSIÓN DE RESULTADOS. CARACTERIZACIÓN DE LA DISTRIBUCIÓN DE FRECUENCIAS ANGULARES.....	202
5.7	EFFECTOS DEL RUIDO DE LAS MEZCLAS EN LA CONVERGENCIA DEL PROCEDIMIENTO PROPUESTO.....	212
5.8	CONCLUSIONES	222

CAPÍTULO 6: CONCLUSIONES, APORTACIONES Y LÍNEAS FUTURAS DE INVESTIGACIÓN..... 223

APÉNDICE I: CONCEPTOS BÁSICOS DE ESTADÍSTICA UTILIZADOS EN LOS MÉTODOS DE SEPARACIÓN DE FUENTES 229

A1.1	INTRODUCCIÓN	230
A1.2	MOMENTOS Y CUMULANTES.....	230
	A1.2.1 Momentos.....	230
	A1.2.2 Cumulantes.....	231
	A1.2.3 Distribuciones de probabilidad multivariantes. Cumulantes cruzados	234
A1.3	FUNCIONES DE DENSIDAD DE PROBABILIDAD DE SEÑALES ALEATORIAS REFERIDAS EN LA PRESENTE MEMORIA	237
	A1.3.1 Señal con densidad de probabilidad uniforme	237
	A1.3.2 Señal con densidad de probabilidad gaussiana (normal)	239
	A1.3.3 Función de densidad de probabilidad gamma	240
	A1.3.4 Función densidad de probabilidad exponencial y doble exponencial (o de Laplace)	244
	A1.3.5 Función densidad de probabilidad beta.....	246

APÉNDICE 2: EL CONCEPTO DE ENTROPÍA EN EL CONTEXTO DE LA SEPARACIÓN DE FUENTES..... 247

A2.1	ENTROPÍA.....	248
A2.2	ENTROPÍA DIFERENCIAL Y VARIABLES ALEATORIAS CONTINUAS	251

A2.3	PRINCIPIO DE MÁXIMA ENTROPÍA	252
A2.4	INFORMACIÓN MUTUA.....	254
A2.5	INFORMACIÓN MUTUA PARA VARIABLES ALEATORIAS CONTINUAS.....	257
A2.6	DIVERGENCIA KULLBACK-LEIBLER.....	258
A2.7	DESCOMPOSICIÓN PITAGÓREA	260

**APÉNDICE 3: RESULTADOS DE SIMULACIÓN OBTENIDOS CON
EL ALGORITMO DE LOCALIZACIÓN DE LOS EJES DE LA
DISTRIBUCIÓN DE VECTORES DE MEZCLA..... 263**

A3.1	RESULTADOS OBTENIDOS CON LA MATRIZ A_1	264
A3.2	RESULTADOS OBTENIDOS CON LA MATRIZ A_2	266
A3.3	RESULTADOS OBTENIDOS CON LA MATRIZ A_3	269
A3.4	RESULTADOS OBTENIDOS CON LA MATRIZ A_4	272
A3.5	RESULTADOS OBTENIDOS CON LA MATRIZ A_5	276
A3.6	RESULTADOS OBTENIDOS CON LA MATRIZ A_6	279
A3.7	RESULTADOS OBTENIDOS CON LA MATRIZ A_7	282
A3.8	RESULTADOS OBTENIDOS CON LA MATRIZ A_8	285

REFERENCIAS BIBLIOGRÁFICAS 289

CAPÍTULO 1:

REVISIÓN DE LOS MÉTODOS DE SEPARACIÓN CIEGA DE FUENTES

En este capítulo se realiza una introducción a la separación ciega de fuentes. En primer lugar (Sec.1.1 y 1.2) se presenta un planteamiento general del problema, indicando los distintos modelos que se suelen considerar, las indeterminaciones inherentes a la resolución del problema y las hipótesis de las que habitualmente se parte.

En la Sec.1.3 se incluyen los principios de los que nacen las distintas propuestas realizadas hasta la fecha para solucionar el problema de separación. De acuerdo con estos principios se pueden agrupar los distintos métodos publicados, y de esta forma se describen a lo largo de las secciones 1.6 a 1.12. Haciendo una abstracción de los trabajos publicados en el campo, la Sec.1.4 presenta las arquitecturas básicas que implementan los distintos métodos de reconstrucción de las fuentes.

Una etapa previa habitual incluida en una gran parte de los algoritmos de separación tiene por objeto efectuar un blanqueado espacial de las señales captadas, que trata de

decorrelacionarlas. En la Sec.1.5 se trata de este proceso, así como del concepto de Análisis de Componentes Principales (PCA), en que suele basarse.

En las secciones 1.6 a 1.12 se resumen los procedimientos y algoritmos más relevantes y conocidos para la separación de fuentes, que parten de un modelo lineal de mezcla instantánea.

El capítulo concluye con una sección dedicada a enumerar las principales aplicaciones de la separación ciega de fuentes (Sec.1.13), y otra sección sobre las aportaciones realizadas en modelos más complejos que el lineal instantáneo (Sec.1.14). Esta última sección da una clara idea de las líneas de investigación abiertas en la actualidad en el campo de la separación ciega de fuentes.

En los Apéndices 1 y 2 se incluyen las bases de Estadística y de Teoría de la información necesarias para entender adecuadamente este capítulo.

1.1 PLANTEAMIENTO/DESCRIPCIÓN DEL PROBLEMA

El problema de separación ciega de fuentes consiste en la obtención de señales generadas por p fuentes, s_j , $j=1, \dots, p$, a partir de las mezclas detectadas por q sensores, e_i , $i=1, \dots, q$. La mezcla de señales tiene lugar en el medio en el que se propagan (Figura 1.1), y:

$$e_i(t) = F_i(s_1(t), \dots, s_j(t), \dots, s_p(t)) \quad i = 1, \dots, q \quad (1.1)$$

donde $F_i: \mathcal{R}^p \rightarrow \mathcal{R}$ es una función de p variables del espacio s al espacio e . En notación vectorial:

$$e(t) = F(s(t)) \quad (1.2)$$

donde:

$$e(t) = (e_i(t))^T \quad s(t) = (s_i(t))^T \quad F = (F_i)^T$$

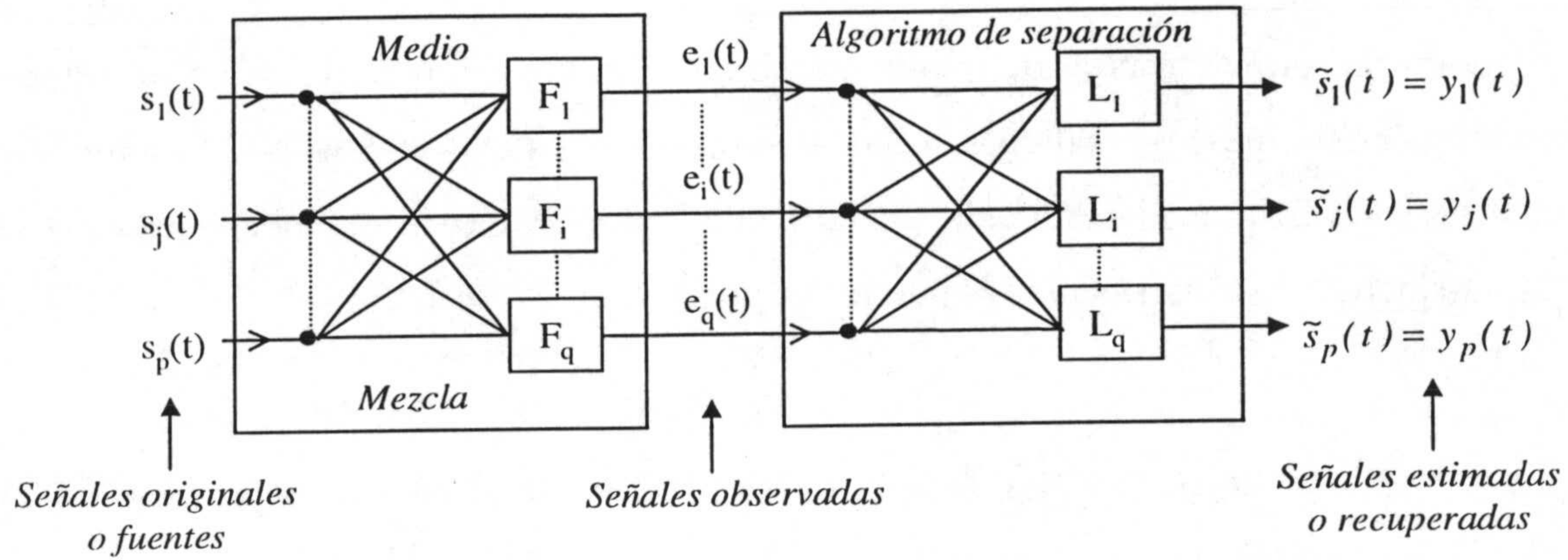


Figura 1.1 Modelo general de mezcla y separación de señales.

La finalidad de la separación de fuentes es obtener p funciones L_j , tales que:

$$s_j(t) = L_j(e_1(t), \dots, e_i(t), \dots, e_q(t)) \quad j = 1, \dots, p \quad (1.3)$$

donde $L_j: \mathcal{R}^q \rightarrow \mathcal{R}$ es una función de q variables del espacio e al espacio s . En notación vectorial:

$$s(t) = L(e(t)) \quad (1.4)$$

donde:

$$L = (L_j)^T$$

L , representa el conjunto de transformaciones, L_j , y es la inversa del conjunto F de transformaciones F_i .

El modelo no lineal representado por (1.1) y (1.2) ha sido estudiado por diversos autores [ALM99], [PUN99], pero es demasiado general y con frecuencia [TAL97, 98], [PUN98b] se considera un modelo post-no lineal (PNL) en el que la señal se propaga a través de un canal de transmisión lineal tras el cual los sensores introducen no linealidades. Por tanto, (1.1) se puede expresar:

$$e_i(t) = F_i \left(\sum_{j=1}^p a_{ij} s_j(t) \right) \quad (1.5)$$

Un modelo más concreto considera que el medio de propagación actúa como un filtro con p entradas: las fuentes, y q salidas: las señales detectadas por los sensores. Usualmente se admite que este filtrado, introducido por el medio y los sensores, es lineal y estacionario. De esta manera las señales captadas pueden expresarse de la forma:

$$e_i(t) = \sum_{j=1}^p \int a_{ij}(t-\tau) \cdot s_j(\tau) \cdot d\tau + n_i(t) \quad \forall i = 1, \dots, q \quad (1.6)$$

En esta expresión $a_{ij}(t)$ representa la respuesta a un impulso de un filtro que modela la propagación entre la fuente i y el sensor j . El término $n_i(t)$ corresponde a un ruido aditivo.

En el dominio de la frecuencia, ν , la expresión (1.6) puede escribirse como (ver Figura 1.2):

$$e_i(\nu) = \sum_{j=1}^p A_{ij}(\nu) \cdot s_j(\nu) + n_i(\nu) \quad \forall i = 1, \dots, q \quad (1.7)$$

o en forma vectorial:

$$\mathbf{e}(\nu) = \mathbf{A}(\nu) \cdot \mathbf{s}(\nu) + \mathbf{n}(\nu) \quad (1.8)$$

donde la matriz $\mathbf{A}(\nu)$, de dimensión $p \times p$, contiene todas las funciones de transferencia entre las p fuentes y los q sensores.

Se puede realizar una simplificación adicional consistente en que la propagación no introduce ningún retardo temporal en cuyo caso las señales detectadas pueden expresarse como (Figura 1.3):

$$e_i(t) = \sum_{j=1}^p a_{ij} \cdot s_j(t) + n_i(t) \quad \forall i = 1, \dots, q \quad (1.9)$$

o en notación vectorial:

$$\mathbf{e}(t) = \mathbf{A} \cdot \mathbf{s}(t) + \mathbf{n}(t) \quad (1.10)$$

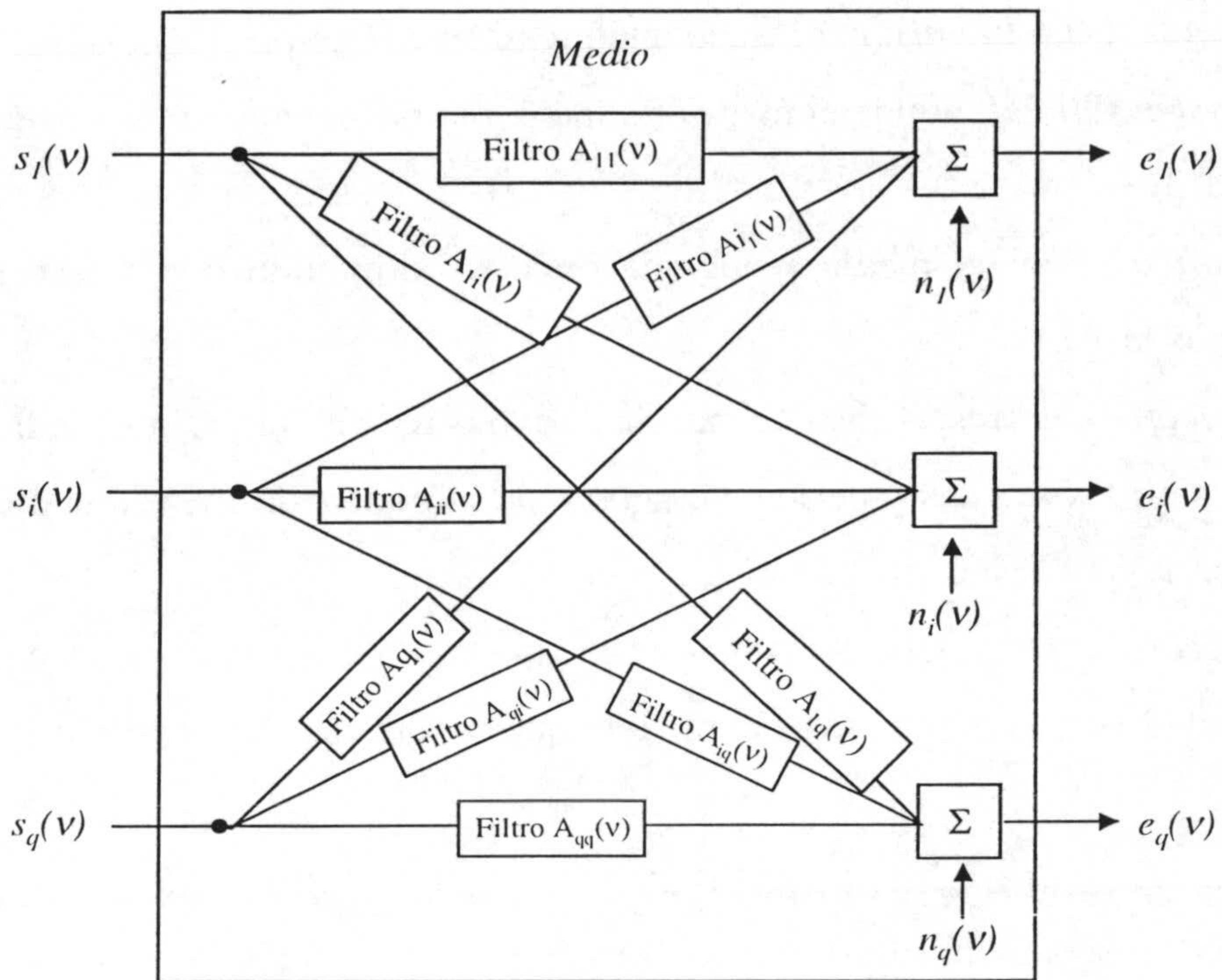


Figura 1.2. Modelo de mezcla convolutiva.

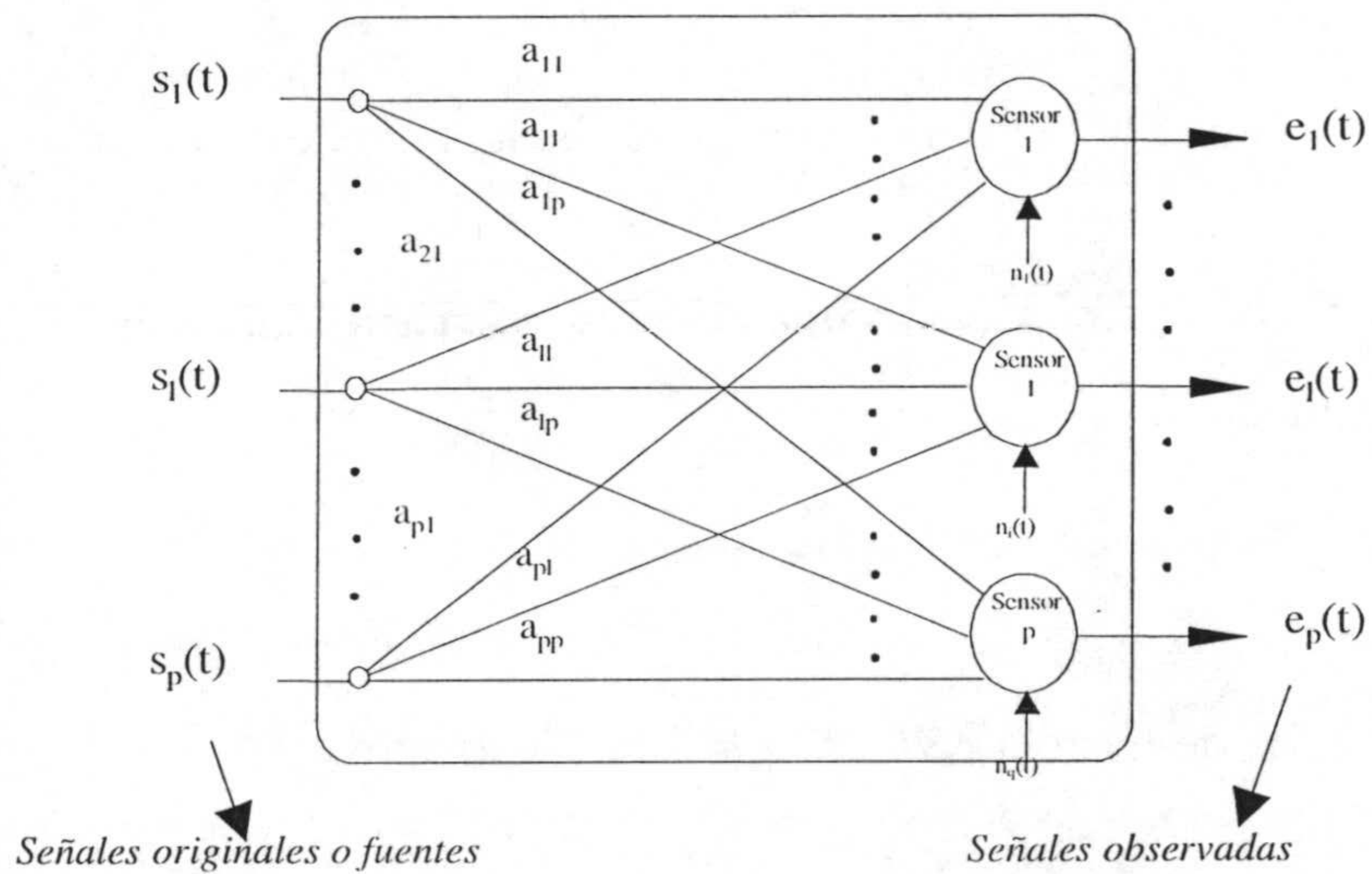


Figura 1.3. Separación ciega de fuentes.

La mezcla obtenida según las expresiones (1.9) y (1.10) se denomina *mezcla lineal*.

En resumen, en el problema de separación de fuentes se suelen considerar los siguientes modelos:

- **Modelo general (no lineal):** expresado analíticamente según (1.1) y (1.2).
- **Modelo post-no lineal:** donde el medio es lineal y la no-linealidad es introducida en los sensores (1.3) y (1.4).
- **Modelo convolutivo:** el medio y los sensores se comportan como filtros lineales y estacionarios (1.6).
- **Mezcla espectral:** corresponde a la mezcla convolutiva en el dominio de la frecuencia. Los valores de las señales y de los elementos de la matriz de mezcla son complejos y vienen descritos por la expresión (1.8).
- **Mezcla lineal:** se considera que tanto el medio como los sensores se comportan de forma lineal (1.9) y (1.10). Este es el modelo que será tratado en la presente memoria.

En las Secs.1.13, 1.14, 2.1.2 se analizará el origen e influencia del ruido en los procedimientos de separación de señales. No obstante, con frecuencia se consideran las mezclas sin ruido de forma que, por ejemplo, para el caso lineal se tiene que:

$$e(t) = A \cdot s(t) \quad (1.11)$$

La resolución del problema de separación ciega de fuentes puede enunciarse así:

Dadas M observaciones independientes del vector de mezcla $e(t)$, encontrar una estimación:

$$\overset{def}{W} = A \quad (1.12)$$

de la matriz de mezcla A . Con la inversa de esta matriz:

$$\overset{def}{B} = W^{-1} \quad (1.13)$$

se podrán recuperar (estimar) las señales originales:

$$\overset{def}{y} = s = W^{-1} \cdot e = B \cdot e \quad (1.14)$$

Los algoritmos de separación de fuentes tratan de explotar principalmente la *diversidad espacial* producida por la separación de los distintos sensores, lo que hace que cada uno de ellos capte, en cada instante de tiempo, distintas mezclas de las fuentes. Por otra parte, los modelos convolutivos tratan de aprovechar la diversidad espectral, aunque el enfoque fundamental de la separación de fuentes es esencialmente espacial y busca la estructura a través de los sensores, no a lo largo del tiempo [CAR98].

1.2 INDETERMINACIONES E HIPÓTESIS

En el presente trabajo se va a considerar únicamente el caso de mezclas lineales e instantáneas, modeladas, por tanto, con la expresión (1.10) o (1.11), dependiendo de si se considera o no, respectivamente, la influencia del ruido.

En la expresión (1.11) se pretende estimar las fuentes, $s(t)$, conociendo tan solo las observaciones, $e(t)$.

La forma de la relación (1.11) y particularmente los valores de $e(t)$ no se modifican si se multiplica cualquiera de las componentes, s_j , de $s(t)$ por una constante, c_j , y si simultáneamente se divide la columna correspondiente de \mathbf{A} , $\mathbf{a}_j=(a_{1j}, \dots, a_{pj})^T$, por esa misma constante. En consecuencia, existe una indeterminación sobre la amplitud o potencia de las fuentes, que no es identificable. En otras palabras, los métodos de separación ciega de fuentes se consideran válidos aunque las señales recuperadas tengan sus amplitudes atenuadas o amplificadas respecto a las fuentes originales.

Otra indeterminación intrínseca al problema concierne al orden de las fuentes. En efecto, en la expresión (1.11) se puede permutar el orden de las componentes de $s(t)$ y simultáneamente y de igual forma se pueden permutar las columnas correspondientes de la matriz \mathbf{A} , sin que por ello varíe el orden de las componentes de las observaciones $e(t)$. Como los procedimientos de separación sólo parten del conocimiento de $e(t)$, resulta imposible obtener la estimación de las fuentes en el mismo orden que las originales. Por lo anterior, no debe extrañar (ver Capítulos 4 y 5) que con frecuencia las señales recuperadas aparezcan en orden distinto a las señales originales.

En resumen, la resolución del problema de separación de fuentes se considera totalmente correcta aunque se recuperen las fuentes con distinta escala de amplitudes y en orden distinto con respecto a las señales originales.

En cuanto a la obtención de la matriz $\mathbf{W} = \tilde{\mathbf{A}}$, las dos indeterminaciones anteriores implican que \mathbf{W} es válida para recuperar las fuentes, aunque sus columnas estén en distinto orden y multiplicadas por constantes arbitrarias, respecto a las columnas de la matriz original de mezcla \mathbf{A} .

Las dos indeterminaciones indicadas (de escala y de orden) se formalizarán matemáticamente en la Sec.2.1.1.

Una vez obtenida la matriz \mathbf{W} , las fuentes se pueden calcular por medio de la expresión (1.14), lo que implica que la matriz \mathbf{W} (o sea, la matriz de mezcla \mathbf{A}) debe ser invertible (no singular). En general, los algoritmos de separación no actúan adecuadamente si \mathbf{A} está *mal condicionada* (es decir, es casi singular). Una primera hipótesis de los métodos de separación es que la matriz que modela la mezcla es invertible.

Una segunda hipótesis se refiere a que el número de observaciones, q , debe ser igual o mayor que el número de fuentes, p ; es decir:

$$q \geq p \quad (1.15)$$

Esta hipótesis es necesaria ya que si no se tendrían más incógnitas (los $p \times p$ elementos de la matriz \mathbf{A} o \mathbf{W}) que ecuaciones (las q observaciones) en (1.11) y (1.14). Por lo general se considera $q=p$, existiendo procedimientos como el de Análisis de Componentes Principales (PCA, ver Sec.1.5) en los que se seleccionan las p componentes de $e(t)$ más adecuadas para la separación caso de que $q > p$. Este problema se volverá a tratar en la Sec.1.14.

La mayoría de los procedimientos de separación de señales son de tipo estadístico (Sec.1.3) y se basan en una tercera hipótesis consistente en que las fuentes son

estadísticamente independientes. No ocurre esto con los procedimientos geométricos, a los que se refiere la presente memoria, que no parten de la independencia estadística. En la Sec.2.1.1 se concretarán las hipótesis que se utilizan en los métodos geométricos.

En resumen, las tres hipótesis que usualmente se suelen realizar en los métodos de separación ciega de fuentes son:

H1.La matriz de mezcla debe verificar: $a_{ij} \neq 0 \forall i \in \{1, \dots, p\}$.

H2.El número de observaciones es igual o mayor que el número de fuentes, $q \geq p$.

H3.Las fuentes son estadísticamente independientes dos a dos.

1.3 PRINCIPIOS DE SEPARACIÓN

Los trabajos presentados hasta ahora para resolver el problema de la separación ciega de fuentes se abordan bajo uno de los siguientes enfoques:

1. *Enfoque estadístico:*

Este enfoque parte de la hipótesis de que las señales originales (o fuentes, $s(t)$) son estadísticamente independientes. Los procedimientos que siguen este enfoque se basan en propiedades de las distribuciones de probabilidad de las fuentes y de las observaciones, caracterizadas por las distribuciones respectivas o por sus cumulantes. Se pretende que las señales recuperadas sean tales que cumplan un determinado criterio estadístico o basado en la Teoría Estadística de la Información. Si las señales originales, $s(t)$, son estadísticamente independientes, las señales recuperadas, $y(t)$, que tratan de estimar las fuentes ($y(t) = \tilde{s}(t)$) también deben ser estadísticamente independientes.

2. *Enfoque geométrico:*

Trata de explotar las propiedades geométricas y algebraicas de las señales originales y de las observaciones, considerando el tipo de transformación realizada por la matriz de mezcla. Este enfoque ha sido propuesto originalmente por investigadores del Departamento de Arquitectura y Tecnología de Computadores de la Universidad de

Granada, y dentro de él se sitúa la presente tesis [PUN95a, 95b, 95c, 95d, 96a, 96b, 96c, 97a, 97b, 98a, 98b, 99], [PRI97, 98, 99a, 99b].

El enfoque geométrico será presentado en el Capítulo 2 de esta memoria. El enfoque primero, basado en la independencia estadística de las fuentes, se analiza en el resto del presente capítulo.

Una gran parte de los trabajos con enfoque estadístico dividen el problema en dos etapas (Figura 1.4.). En la primera se transforman las señales de entrada, $e(t)$, de forma que resulten, $x(t)$, con media cero, decorrelacionadas, y con varianza unidad. Es decir, la transformación V :

$$x(t) = Ve(t) \quad (1.16)$$

debe ser tal que:

$$\langle x(t) \rangle = 0; \quad \langle x(t)x^T(t) \rangle = I \quad (1.17)$$

con ello, las covarianzas son nulas: $\langle x_i x_j \rangle = 0$, si $i \neq j$, y las varianzas de cada señal son la unidad $\langle x_i^2 \rangle = 1$, $\forall i$.

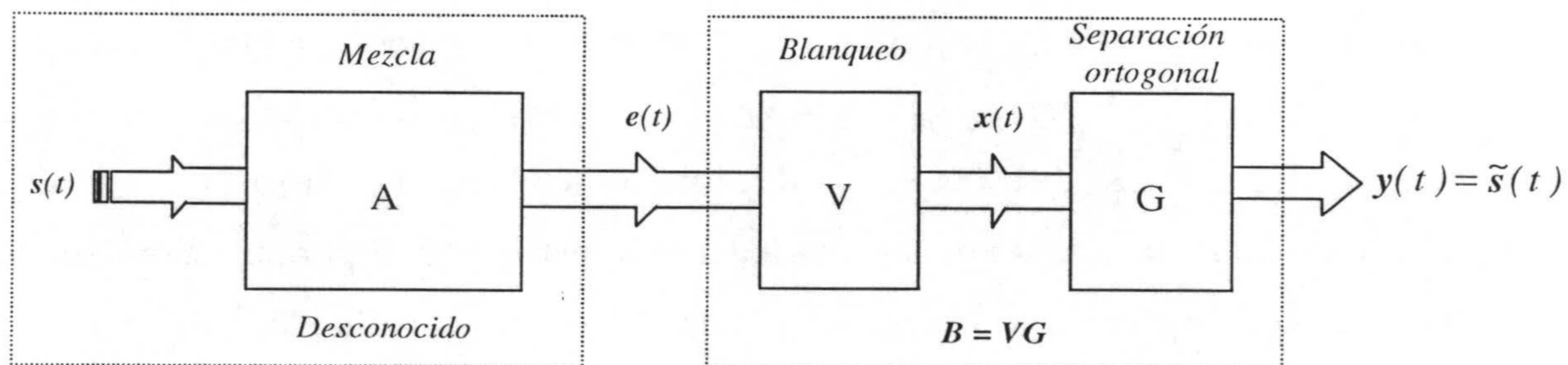


Figura 1.4. Problema de separación de fuentes en dos etapas.

Con esta transformación las señales están decorrelacionadas; dicha condición es necesaria pero no suficiente para la independencia estadística (ver Sec.A1.2.3), por lo tanto, es necesaria una transformación adicional (G , en la Figura 1.4.) para obtener la separación.

Se demuestra [LAC97, CAR98, CAR99] que si las señales $x(t)$ están blanqueadas, la matriz G de separación puede ser ortonormal, es decir:

$$\mathbf{G}\mathbf{G}^T = \mathbf{I} \quad (1.18)$$

con lo que se facilita notablemente el proceso de separación.

La Sec.1.5 se dedicará a analizar con más detalle el proceso de blanqueado de señales.

1.3.1. Criterios estadísticos de separación. Funciones de contraste.

La separación de fuentes basada en propiedades estadísticas se suele obtener optimizando una función de contraste; es decir, optimizando una medida escalar de alguna “propiedad de la distribución” de las salidas, $y(t)$. Estas funciones de contraste se pueden basar en: la minimización de la información mutua, la prueba de máxima verosimilitud, el principio de la máxima entropía, en decorrelaciones de alto orden, etc.

Las funciones de contraste son funciones objetivo, y con ellas se efectúa una *estimación de contraste mínimo*, que es una técnica general de inferencia estadística. En este capítulo se designarán con la forma $\Phi[y]$, la aplicación de la función de contraste a la función de distribución y . Usualmente se incluirá un subíndice que indicará el criterio estadístico utilizado. Por ejemplo, $\Phi_{\text{IM}}[y]$ se refiere a una función de contraste basada en el principio de maximización de la información (IM: “Information Maximization”).

Caso de que se aplique a señales previamente blanqueadas diremos que es una *función de contraste ortogonal*, y se designará como $\Phi_{\text{IM}}^{\circ}[y]$.

Las funciones de contraste $\Phi[y]$ deben elegirse de forma que la separación se obtenga cuando alcance un valor mínimo. En otras palabras, una función de contraste, $\Phi[y]$, es adecuada si:

$$\Phi[y] = \Phi[\mathbf{B}\cdot\mathbf{A}\cdot s] \geq \Phi[s] \quad (1.19)$$

verificándose la igualdad sólo cuando $y = \mathbf{B}\cdot\mathbf{A}\cdot s$ es una copia de las señales originales s .

En el caso de una función de contraste ortogonal, debe minimizarse $\Phi^0[\mathbf{y}]$ con la condición de blanqueo, $E\{\mathbf{y}\mathbf{y}^T\}=\mathbf{I}$.

En las Sec.1.6 a 1.11 se obtendrán distintas funciones de contraste de acuerdo con los siguientes criterios (Figura 1.5):

- a) Mínima información mutua entre las señales de salida (Figura 1.5a). Conduce a las funciones de contraste Φ_{MI} y Φ_{ICA} (Sec.1.7).
- b) Minimización de la divergencia entre las distribuciones de salida, $f[\mathbf{y}]$, y las del modelo de fuentes, $f[s]$. Se obtiene la función de contraste de máxima verosimilitud, Φ_{ML} (Figura 1.5b), (Sec.1.8).
- c) La mezcla reduce la entropía de las fuentes, supuestas éstas estadísticamente independientes. El criterio consiste en maximizar la entropía de las salidas (Φ_{EM}) (Figura 1.5c).
- d) Las funciones de distribución se pueden caracterizar adecuadamente con sus cumulantes (Apéndice 1). Para que las señales de salida estimen a las fuentes, se minimiza la distancia entre los cumulantes de salida y los cumulantes del modelo de fuentes (Figura 1.5d) (Sec.1.10).
- e) Minimizar el error cuadrático medio entre las señales observadas, e_k , y una reproducción de dichas señales (Figura 1.5e). El método se denomina análisis de componentes principales no lineal (PCA, Sec.1.11) y la función de contraste se denominará Φ_{PCAnl} .

Diversos autores [CAR98, CAR99, HAY99, AMA97] han mostrado la conexión existente entre unos y otros procedimientos (ver, por ejemplo Sec.1.8).

En las Sec.1.6 a 1.11 se analizan los distintos criterios obteniéndose diferentes funciones de contraste.

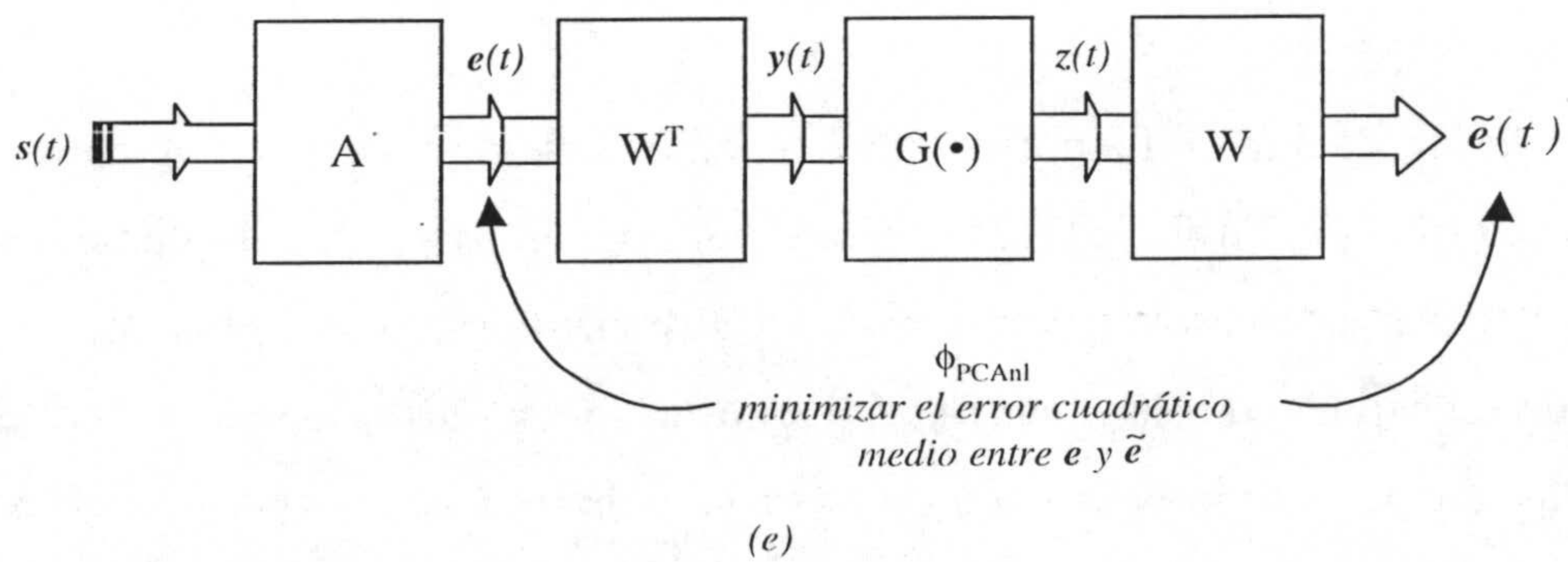
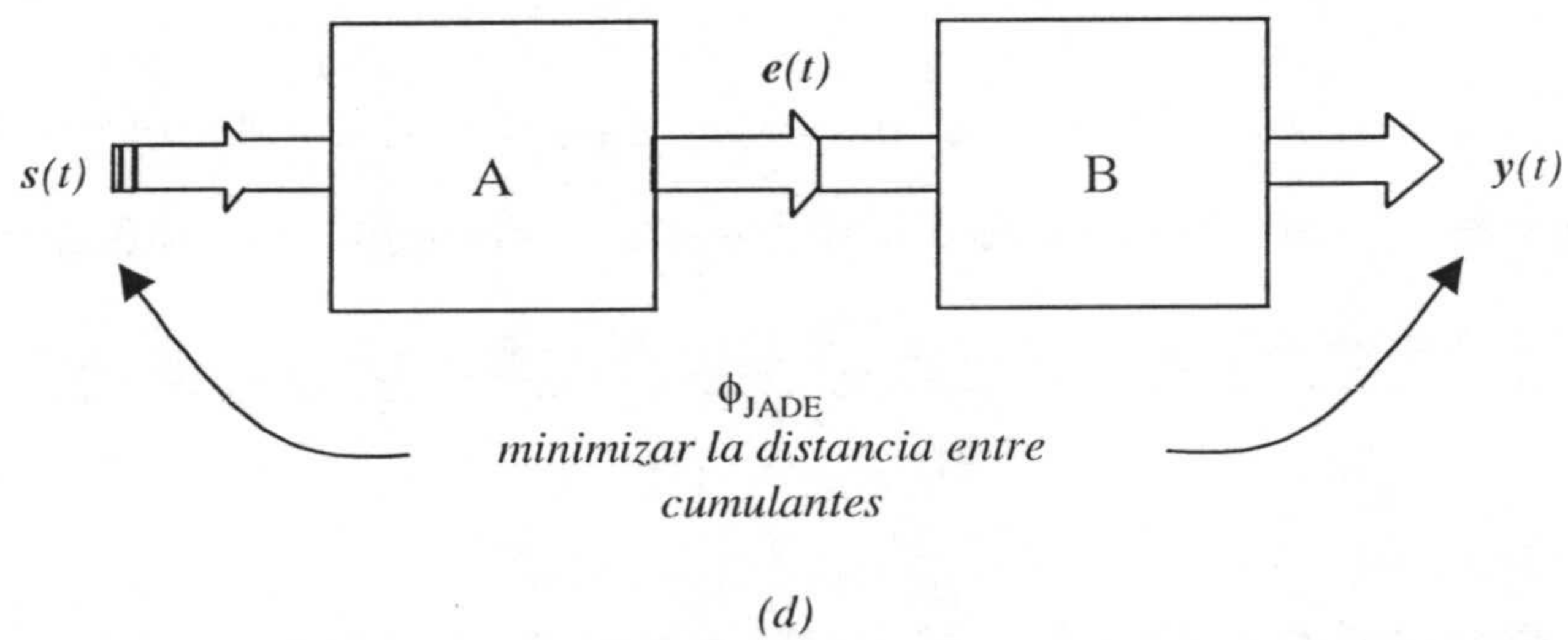
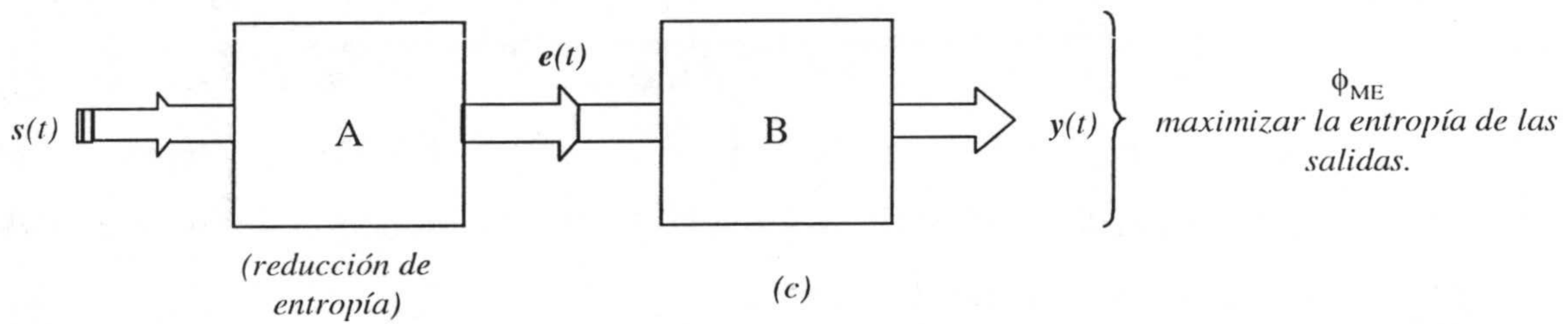
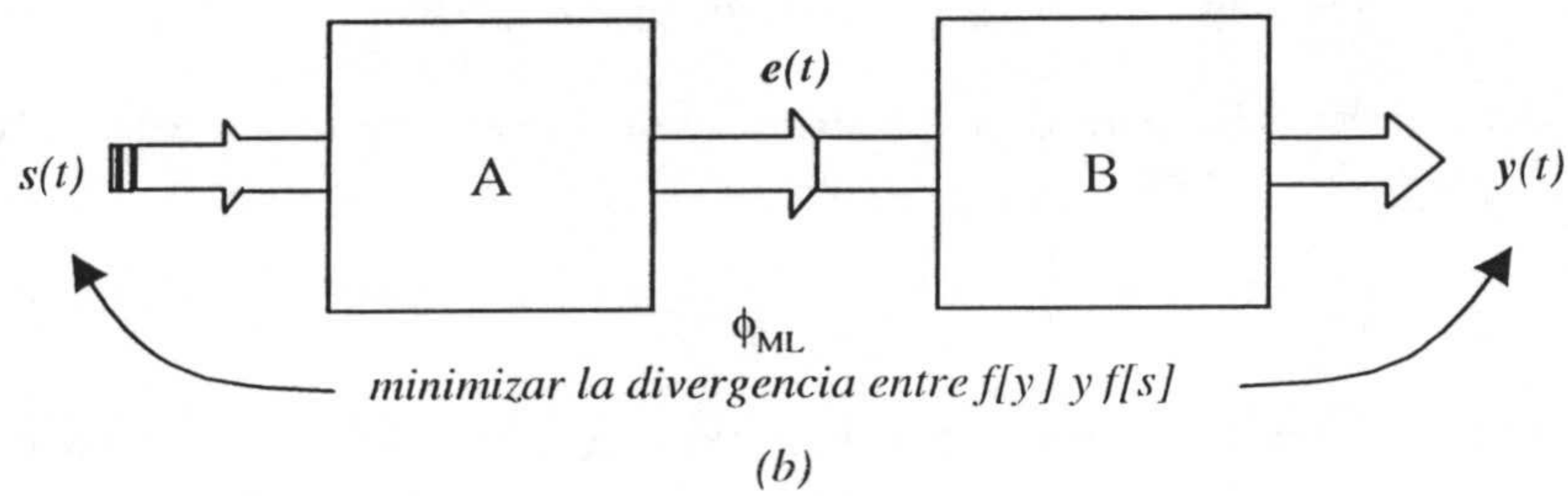
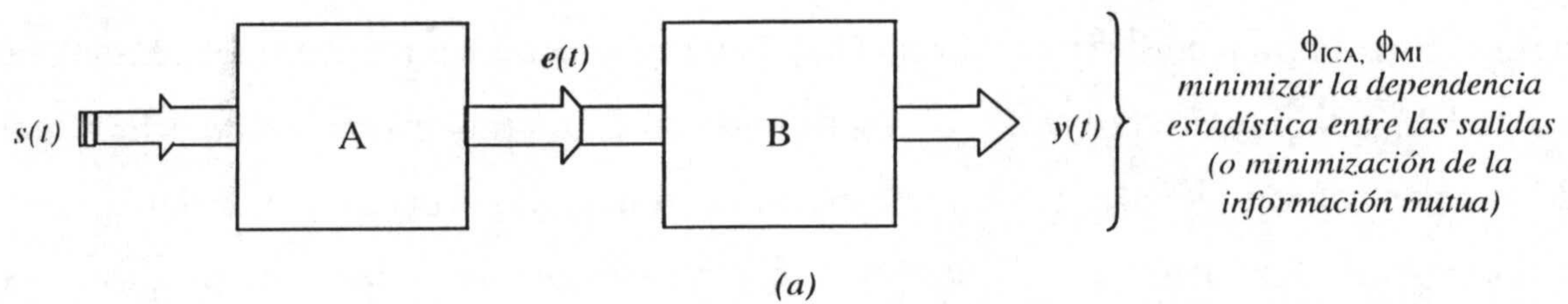


Figura 1.5. Criterios para la separación de fuentes.

1.3.2. Análisis de Componentes Independientes (ICA).

El análisis de Componentes Principales (PCA) es una técnica matemática (que se analizará en la Sec.1.5) que permite proyectar un *espacio de datos* en un *espacio de características* donde los ejes son ortogonales (como el espacio original), de forma que la varianza de las proyecciones de los datos en dichos ejes es mayor que en cualesquiera otros ejes ortogonales posibles. Para realizar esta transformación se impone la condición de que las nuevas señales estén decorrelacionadas (es decir, las covarianzas o cumulantes de 2º orden deben ser 0).

El concepto de Análisis de Componentes Independientes (ICA, Independent Component Analysis) fue introducido por Jutten y Héroult [JUT88], y Comon [COM94], como una ampliación del PCA. Mientras el PCA sólo impone la condición de la independencia estadística de segundo orden y de que la base de vectores del espacio de características sea ortogonal, ICA impone la *independencia estadística* de las componentes individuales del vector de salida y sin la condición de ortogonalidad.

Con frecuencia se asocia el término de ICA con el de *separación ciega de fuentes*, en general. Rigurosamente hablando ICA corresponde únicamente al criterio a) citado en la Sec.1.3.1 (Figura 1.5a).

1.4 ARQUITECTURAS PARA SEPARACIÓN DE FUENTES.

Un sistema de separación de fuentes puede implementarse de forma analógica funcionando en forma continua [VIT89], [COH92]. Lo usual, sin embargo, es implementarlo en un sistema digital. En este caso, la captación de las señales, $e(t)$, se efectúa con conversores A/D situados después de los sensores. De esta forma las señales a procesar están discretizadas en el tiempo, $e(m)$, con $t = mT_s$ y $m=0,1,\dots$, siendo T_s el periodo de muestreo.

Hay dos formas de implementar los algoritmos de separación de señales: por medio de sistemas directos y por medio de sistemas realimentados. En los *sistemas directos* (Figura

1.6a) los parámetros del separador (elementos b_{ij} de la matriz \mathbf{B}) se calculan directamente a partir de las observaciones $e(t)$. De esta forma se suelen obtener los parámetros de la transformación de blanqueo espacial \mathbf{V} (Figura 1.4.).

Otro tipo de algoritmos se implementan con *sistemas realimentados*. En ellos los parámetros del separador para un instante de tiempo $m+1$, $\mathbf{B}(m+1)$, se obtienen a partir de los valores provisionales de las fuentes estimadas, $y(m)$, y de las observaciones, $e(m)$, en el instante presente m .

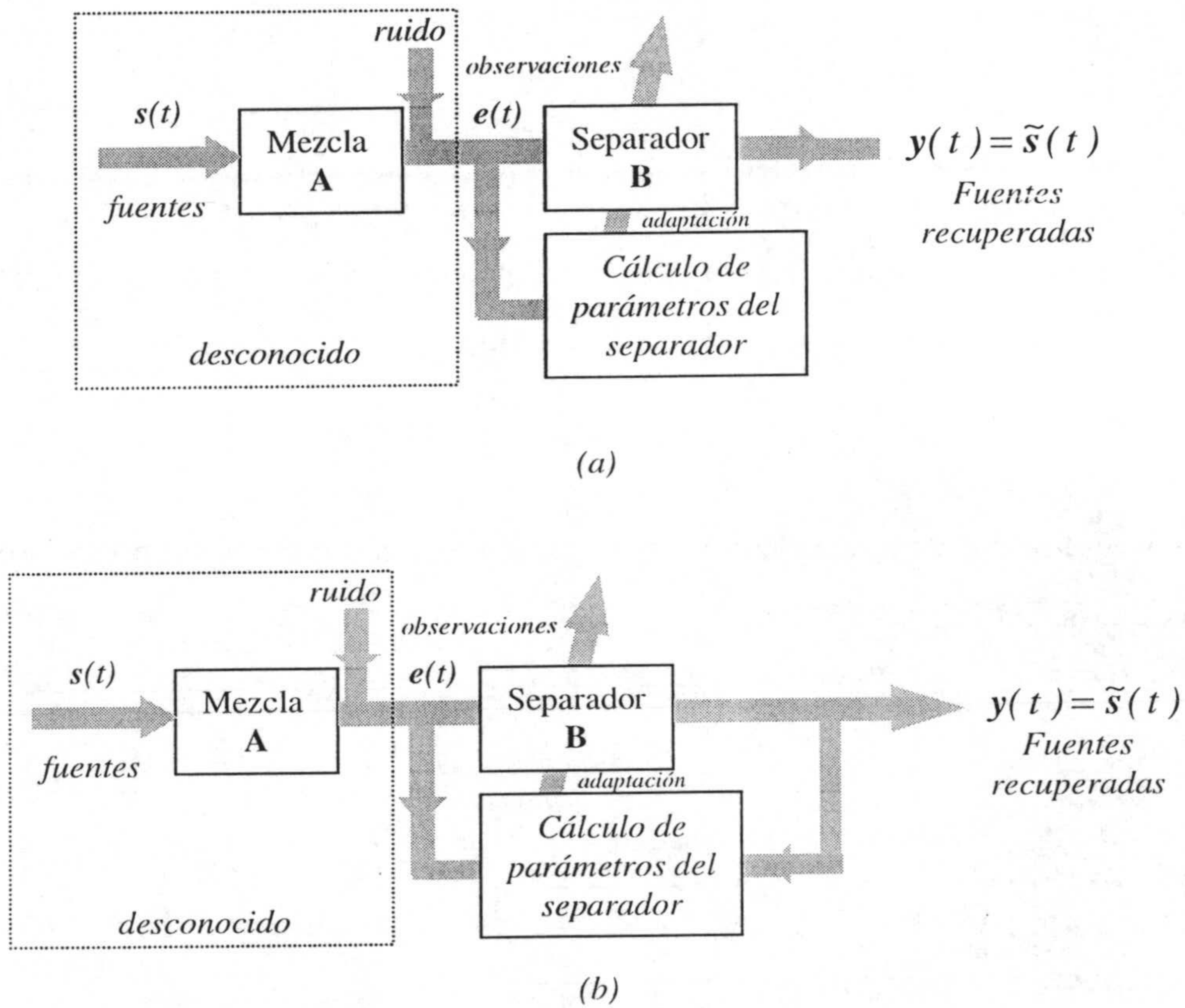


Figura 1.6. Implementaciones de algoritmos de separación.

Si la actualización de \mathbf{B} se hace a partir de un bloque o serie de muestras consideradas globalmente, se dice que el sistema o algoritmo actúa *fuera de línea* o *por bloques* ("batch"). Si por el contrario la adaptación de los parámetros del separador se realiza muestra a muestra, se dice que el algoritmo actúa *en línea*. Caso de que la actualización de los parámetros y la generación de nuevas salidas se realice en un tiempo t , menor que el periodo de muestreo, $t < T_s$, se dice que el sistema actúa en *tiempo real*.

Con frecuencia, el separador se implementa con una red neuronal artificial (ANN, “Artificial Neural Network”) (Figura 1.7). En este caso en vez de denominar \mathbf{B} a la matriz de parámetros del separador se denomina \mathbf{W} (por utilizar la terminología usada en el campo de las ANNs), y los elementos del separador, w_{ij} , se denominarán *pesos sinápticos* o *pesos* sencillamente [PRI96]. La adaptación de pesos w_{ij} de la red neuronal se realiza por medio de un algoritmo adaptativo de aprendizaje, como se verá a continuación.

El aprendizaje o entrenamiento de la red supone la modificación de los pesos de la misma:

$$w_{ij}(m+1) = w_{ij}(m) + \Delta w_{ij}(m) \quad (1.20)$$

y suele efectuarse mediante el método de *descenso del gradiente*. Este es el caso, por ejemplo, del método de minimización de la información mutua, que se analizará en la Sec.1.7. El ajuste de los pesos puede expresarse como:

$$\Delta w_{ij} = -\eta \frac{\partial \phi}{\partial w_{ij}} \quad (1.21)$$

donde η es la *ganancia de aprendizaje* y ϕ es la función de contraste expresada en función de los pesos de la red.

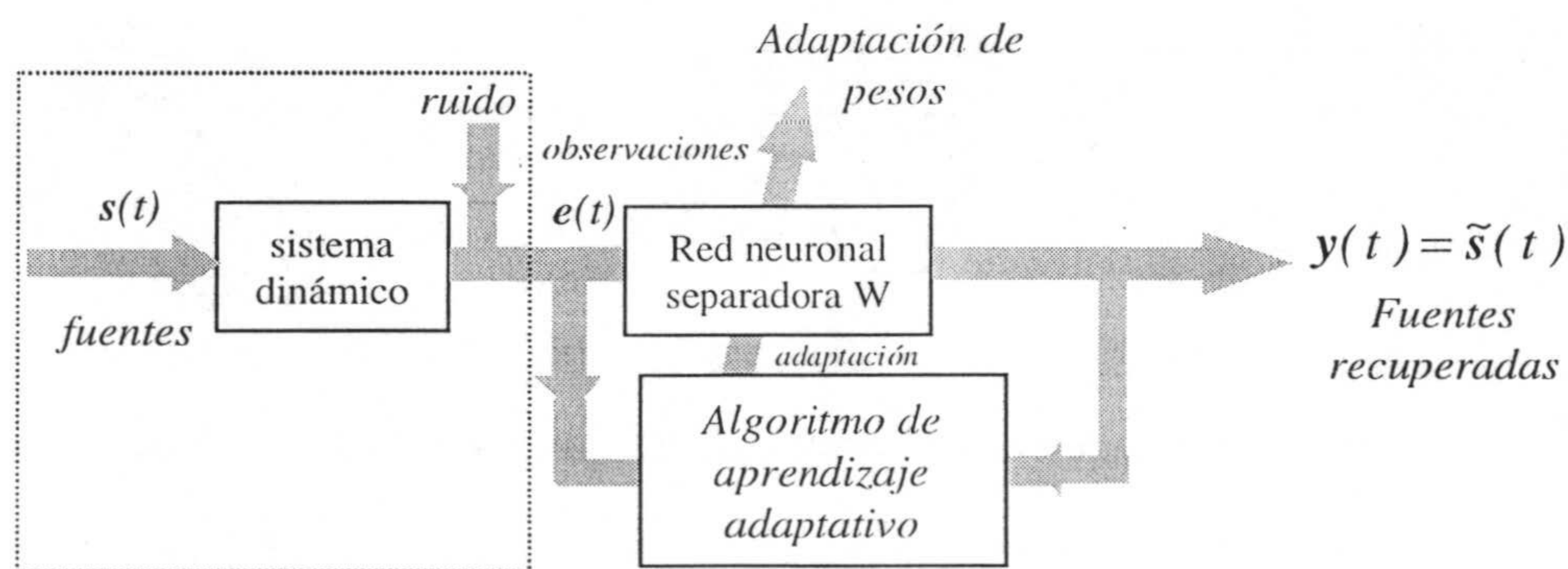


Figura 1.7 Criterio para la separación de fuentes.

La expresión (1.21) tiene un claro significado. ϕ representa una hipersuperficie en el espacio de los pxp pesos. En cada instante de tiempo (m) habrá definidos unos valores concretos de los pesos $w_{ij}(m)$ y por tanto estaremos en un punto concreto $P = \phi[w(m)]$ de la hipersuperficie.

Según se indicó en la Sec.1.3.1 los valores de \mathbf{W} serán correctos (es decir, producirán la separación) cuando el valor de la función de contraste ϕ sea mínimo. Entonces el algoritmo de aprendizaje debe ser tal que lleve los pesos a buscar un mínimo de la superficie ϕ . Es lo que hace la expresión (1.21), ya que indica que hay que modificar los pesos en una dirección tal que dicho cambio reduzca el valor de ϕ ; es decir, de forma que el punto de la superficie vaya descendiendo (*gradiente negativo*) hacia un mínimo.

Cuando P se encuentre en un mínimo habremos logrado el objetivo (los w_{ij} son tales que ϕ es mínimo) y $\partial\phi/\partial w_{ij}$ será cero, no volviéndose a actualizar los pesos ($\Delta w_{ij}=0$ según (1.21)).

Existen otros algoritmos de aprendizaje de redes neuronales que también se usan en el contexto de separación ciega de fuentes. Por ejemplo, en el método de máxima entropía [BEL95a, 95b, 96] (Sec.1.9) se utiliza el de *ascenso más abrupto*, que viene descrito por:

$$\Delta W = \eta \frac{\partial \phi}{\partial W} \quad (1.22)$$

En el algoritmo de análisis de componentes principales (PCA) se usa un algoritmo hebbiano generalizado (GHA, "Generalized Hebbian Algorithm") [HAY99], con el que el cambio Δw_{ij} aplicado al peso w_{ij} en cada iteración m es:

$$\Delta w_{ij}(m) = \eta \left[y_i(m) \cdot e_j(m) - y_j(m) \sum_{k=1}^j [w_{ki}(m) y_k(m)] \right] \quad (1.23)$$

Se denomina hebbiano porque se refuerzan los pesos cuyos extremos (entrada e_i , salida y_i) estén simultáneamente activos. Es decir, el incremento Δw_{ij} es proporcional al *producto* $y_i \cdot e_i$, cosa que ocurre con el primer término del segundo miembro de (1.23).

El gradiente utilizado en la expresión (1.21) proporciona la dirección óptima en el espacio de pesos de descenso de ϕ hacia un mínimo sólo cuando el espacio de parámetros es euclídeo en un sistema de coordenadas ortonormal. Sin embargo, con frecuencia (sobre todo en las implementaciones neuronales) el sistema de coordenadas no es ortonormal. En este caso es más conveniente utilizar como gradiente:

$$\nabla^* \phi = (\nabla \phi) \cdot W^T W \quad (1.24)$$

ya que este provoca un descenso más pronunciado y efectivo.

El gradiente definido en (1.24) fue introducido independientemente por Amari y Cardoso. Amari [AMA96, 98a] lo denominó *gradiente natural* y lo obtuvo desde una perspectiva geométrica (para cualquier modelo estadístico suave), y Cardoso ([CAR98], por ejemplo) lo denominó *gradiente relativo* y lo definió para cualquier grupo continuo de transformaciones. Previamente Cichocki [CIC94] propuso un algoritmo similar.

De (1.21) y (1.24) se deduce que para hacer la actualización de pesos, la función de contraste no puede ser constante ya que entonces siempre sería $\nabla \phi = 0$. ϕ debe tener claramente definido algún mínimo. A veces no ocurre así y para tener una superficie más adecuada se transforman las señales de salida, y_i , con alguna función no lineal, $g(y_i)$. El método del gradiente se aplica a las señales $g(y_i)$ siendo las fuentes recuperadas los valores correspondientes de y_i [BEL95a, 95b, 96], [OJA97].

De acuerdo con (1.20), (1.21) y (1.24), en el caso de utilizar el gradiente natural la actualización de pesos se realiza según la siguiente expresión:

$$W(m+1) = W(m) - \eta W^T(m) \cdot W(m) \frac{\partial \phi(m)}{\partial W} = (I - \eta \cdot H(y(m))) W(m) \quad (1.25)$$

donde se ha definido una función de actualización (*función de estimación*), $H(y(m))$ de la siguiente forma:

$$H(y(m)) \stackrel{def}{=} W^T(m) \cdot \frac{\partial \phi(m)}{\partial W} \quad (1.26)$$

Esta función transforma los vectores $y(m)$ en una matriz $p \times p$; es decir $\mathcal{R}^p \rightarrow \mathcal{R}^{p \times p}$.

La recuperación de señales puede efectuarse aplicando la expresión (1.25). Se logrará el régimen estacionario cuando el valor medio de la función de actualización sea cero [CAR99]; es decir, cuando se verifique la siguiente *ecuación de estimación*:

$$\frac{1}{m} \sum_{l=1}^m H(y(l)) = 0 \quad (1.27)$$

1.5 BLANQUEADO ESPACIAL. ANÁLISIS DE COMPONENTES PRINCIPALES.

Según se indicó en las Sec.1.3 y 1.4 es frecuente efectuar un blanqueado espacial de las señales observadas antes de realizar la separación de señales. El blanqueado consiste en someter a los vectores $e(t)$ a una transformación lineal V de forma que se obtengan unos nuevos vectores $x(t)$ dados por (Figura 1.4.):

$$x(t) = V \cdot e(t) \quad (1.28)$$

de forma que:

$$E\{x(t)x^T(t)\} = I \quad (1.29)$$

La matriz V puede ser estimada a partir de una muestra de los vectores $e(t)$ calculando de alguna forma su matriz de covarianza y después normalizando, utilizando los valores propios (como se verá más adelante en esta sección). También el blanqueado puede realizarse neuronalmente.

Antes de nada se hace que los vectores de entrada $e(t)$ tengan media cero. Esto se logra sin más que restarles su media, es decir:

$$e(t) \leftarrow e(t) - E\{e(t)\} \quad (1.30)$$

De esta forma los datos se normalizan con respecto a la estadística de primer orden. Los efectos de la estadística de 2º orden en las no linealidades pueden eliminarse blanqueando los datos con la transformación (1.28). Las componentes de los vectores blanqueados $x(t)$ resultan estar decorrelacionadas y normalizadas de forma que su varianza es la unidad, como indica la expresión (1.128). Por consiguiente, a partir de las observaciones se obtienen unas señales estandarizadas formadas por componentes decorrelacionadas y de la misma potencia. Según se indicó en la Sec.1.3, la decorrelación es un prerequisite necesario (pero no suficiente) para obtener la independencia estadística.

Existen diversas formas de decorrelacionar los datos de entrada. A continuación se analizan algunas posibilidades.

Con frecuencia los métodos de blanqueo se basan en la realización de un *análisis de componentes principales* (PCA), ya que con él simultáneamente pueden conseguirse diversos objetivos:

- *Comprimir la información*, en el sentido de optimizar el error cuadrático medio. El PCA permite transformar un *espacio de datos* (q -dimensional) en un *espacio de características* (p -dimensional, $p < q$), ya que los datos iniciales pueden quedar representados adecuadamente por características (números) que mantienen la información intrínseca. En el contexto de la separación de fuentes si se desconoce el número p de fuentes y el número de sensores es $q > p$, el PCA puede seleccionar automáticamente las p componentes más relevantes o independientes (Sec.1.14).
- *Filtrar el posible ruido gaussiano contenido en las señales.*
- *Decorrelacionar las señales* con objeto de blanquearlas.
- *Realizar la separación de las fuentes* con un procedimiento *PCA no lineal*, que se describirá en la Sec.1.11, que fue propuesto por Oja [OJA91, 95, 97].

En [HAY99] puede encontrarse un estudio detallado del PCA que se resume a continuación.

El análisis de componentes principales parte de señales $e(t)$ de media nula (1.30) y trata de buscar una matriz ortogonal \mathbf{Q} tal que las componentes $x_i(t)$ de:

$$\mathbf{x}(t) = \mathbf{Q} \cdot \mathbf{e}(t) \quad (1.31)$$

están decorrelacionadas, es decir:

$$\mathbf{E}\{\mathbf{x}(t) \cdot \mathbf{x}^T(t)\} = \mathbf{D} \quad (1.32)$$

donde \mathbf{D} es una matriz diagonal.

Esto puede lograrse fácilmente a partir de la *matriz de correlación* de los datos de entrada:

$$\mathbf{R}_{ee} = \mathbf{E}\{\mathbf{e}(t) \cdot \mathbf{e}^T(t)\} \quad (1.33)$$

Como \mathbf{R}_{ee} es cuadrada ($q \times q$) y simétrica, se verifica:

$$\mathbf{R}_{ee}^T = \mathbf{R}_{ee} \quad (1.34)$$

Debido a que \mathbf{Q} es una transformación lineal la *matriz de correlación* de salida puede expresarse:

$$\mathbf{R}_{xx} = E\{\mathbf{x}(t) \cdot \mathbf{x}^T(t)\} = E\{\mathbf{Q}\mathbf{e}(t) \cdot \mathbf{Q}^T \mathbf{e}^T(t)\} = \mathbf{Q}\mathbf{R}_{ee}\mathbf{Q}^T = \mathbf{D} \quad (1.35)$$

donde la matriz $\mathbf{D}=(d_{ij})$ es diagonal y contiene los p valores propios, λ_i , mayores de \mathbf{R}_{ee} ordenados de mayor a menor ($\lambda_1 > \dots > \lambda_j > \dots > \lambda_p$) y \mathbf{Q} es una matriz cuyas columnas \mathbf{q}_j son los vectores propios de \mathbf{R}_{ee} correspondientes y en el mismo orden que los valores propios, λ_i , incluidos en \mathbf{D} .

La matriz \mathbf{Q} es ortogonal ya que sus columnas (los vectores propios de \mathbf{R}_{ee}) satisfacen la condición de ortonormalidad, es decir:

$$\left. \begin{aligned} \mathbf{q}_i^T \mathbf{q}_j &= 1 & \text{si } i = j \\ \mathbf{q}_i^T \mathbf{q}_j &= 0 & \text{si } i \neq j \end{aligned} \right\} \quad (1.36)$$

con lo que:

$$\mathbf{Q}^T \mathbf{Q} = \mathbf{I} \quad \mathbf{Q}^T = \mathbf{Q}^{-1} \quad (1.37)$$

De acuerdo con la definición de \mathbf{R}_{ee} (1.33) y como los vectores \mathbf{q} son unitarios, se puede escribir:

$$\mathbf{q}_j^T \cdot \mathbf{R}_{ee} \mathbf{q}_k = \sigma_j^2 \quad j = 1, \dots, p_j \quad (1.38)$$

donde σ_j^2 es la varianza de la señal e_j .

Comparando (1.35) con (1.38) se tiene:

$$d_{jj} = \lambda_j = \sigma_j^2 \quad (1.39)$$

De acuerdo con las expresiones anteriores se puede deducir que:

- a) Los vectores propios de la matriz de correlación, \mathbf{R}_{ee} , correspondientes a los vectores de entrada $\mathbf{e}(t)$, de media cero, definen vectores unitarios representando las direcciones principales a lo largo de las cuales las varianzas toman sus valores máximos.

- b) Los valores propios asociados definen los valores máximos de esas varianzas.
- c) Si se considera un número de valores propios y vectores propios p , menor que el número de componentes q , de los vectores de entrada, se consigue una *compresión de datos* ($p < q$). Recuérdese que en la matriz \mathbf{D} se ordenan los valores propios de mayor a menor, con lo que queda garantizado que se seleccionan las componentes principales (los que tienen mayor varianza), y por tanto los de mayor contenido de información. Desde otro punto de vista, la compresión de datos se realiza de forma que el error cuadrático medio resulta ser mínimo.
- d) Proporciona datos, \mathbf{x} , decorrelacionados (1.32).

El análisis de componentes principales puede aplicarse utilizando técnicas de cálculo convencionales o redes neuronales artificiales. Así Sanger [SAN89] propuso una red neuronal auto-organizativa con este objetivo, y que puede entrenarse por medio de un algoritmo hebbiano generalizado (GHA, "Generalized Hebbian Algorithm"). Los detalles pueden verse en la referencia [HAY99].

Según (1.35) el PCA decorrelaciona las señales, pero como $d_{ii} \neq 0$, no quedan blanqueadas (varianza = 1). Para producir el blanqueo, la matriz de transformación entre $e(t)$ y $x(t)$ ($x(t) = \mathbf{V}e(t)$) debe ser:

$$\mathbf{V} = \sqrt{\mathbf{D}^{-1}} \cdot \mathbf{Q}^T \quad (1.40)$$

ya que de esta forma:

$$\mathbf{R}_{xx} = \mathbf{E}\{\mathbf{x} \cdot \mathbf{x}^T\} = \mathbf{I} \quad (1.41)$$

Un algoritmo adaptativo para obtener la matriz de blanqueo ha sido propuesto por [LAH94], [SIL91] y [PLU93], y se resume en la siguiente expresión:

$$\mathbf{V}(m+1) = \mathbf{V}(m) - \eta(m) [\mathbf{x}(m)\mathbf{x}^T(m) - \mathbf{I}] \mathbf{V}_m \quad (1.42)$$

Este algoritmo es muy sencillo pero a diferencia del (1.40) no puede comprimir datos y a veces presenta problemas de estabilidad [OJA97].

Por último, Bell y Sejnowski [BEL95a, 95b] proponen la siguiente matriz de blanqueo en su procedimiento de separación de fuentes basado en la maximización de la entropía:

$$V = 2\sqrt{E^{-1}} \{e \cdot e^T\} \quad (1.43)$$

con la que se obtienen resultados adecuados en las aplicaciones que consideran.

1.6 EL PROCEDIMIENTO PIONERO DE SEPARACIÓN DE FUENTES: ARQUITECTURA NEUROMIMÉTICA DE JUTTEN-HERAULT.

Los primeros trabajos sobre separación de señales y el concepto en si mismo fue introducido a partir de mediados de la década de los 80 por Héault, Jutten y Ans [HER85, HER86, JUT91].

Trataban de modelar el problema biológico de la codificación del movimiento. Observando el comportamiento biológico se puede suponer que los circuitos del sistema nervioso son capaces de recibir información a través de unos receptores y a partir de ella sustraer y transmitir distinta información dedicada cada una a una tarea específica. Es decir, los sistemas biológicos son capaces de extraer información a partir de una mezcla concreta. A partir de estas observaciones y otras consideraciones como la adaptabilidad de los sensores biológicos (evolución debida a lesiones, edad, etc.), dichos autores propusieron una solución basada en una aproximación neuronal cuya arquitectura se parece a la de algunos niveles del sistema nervioso central (nivel sensorial, cortical, cerebral).

Un estudio detallado de los primeros trabajos de Herault y Jutten puede encontrarse en [HER94].

El método de separación de Jutten y Héault propone una red neuronal lineal recursiva, similar a la red de Hopfield, completamente interconectada, con capacidad de aprendizaje, basada en un test de independencia que usa funciones no lineales.

La red neuronal está formada por p neuronas lineales. La salida $y_i(t)$ es una suma ponderada de las señales de entrada $e_i(t)$, y del resto de las salidas $y_k(t)$. A la salida de la red neuronal (Figura 1.8) tenemos:

$$y_i(t) = e_i(t) - \sum_{\substack{k=1 \\ k \neq i}}^p w_{ik} \cdot y_k(t) \quad 1 \leq i \leq p \quad (1.44)$$

donde w_{ik} es la matriz de pesos de la red. Nótese que los coeficientes de la diagonal de w_{ij} son nulos. Esta expresión también se puede escribir de forma matricial:

$$\mathbf{y}(t) = \mathbf{e}(t) - \mathbf{W} \cdot \mathbf{y}(t) \quad (1.45)$$

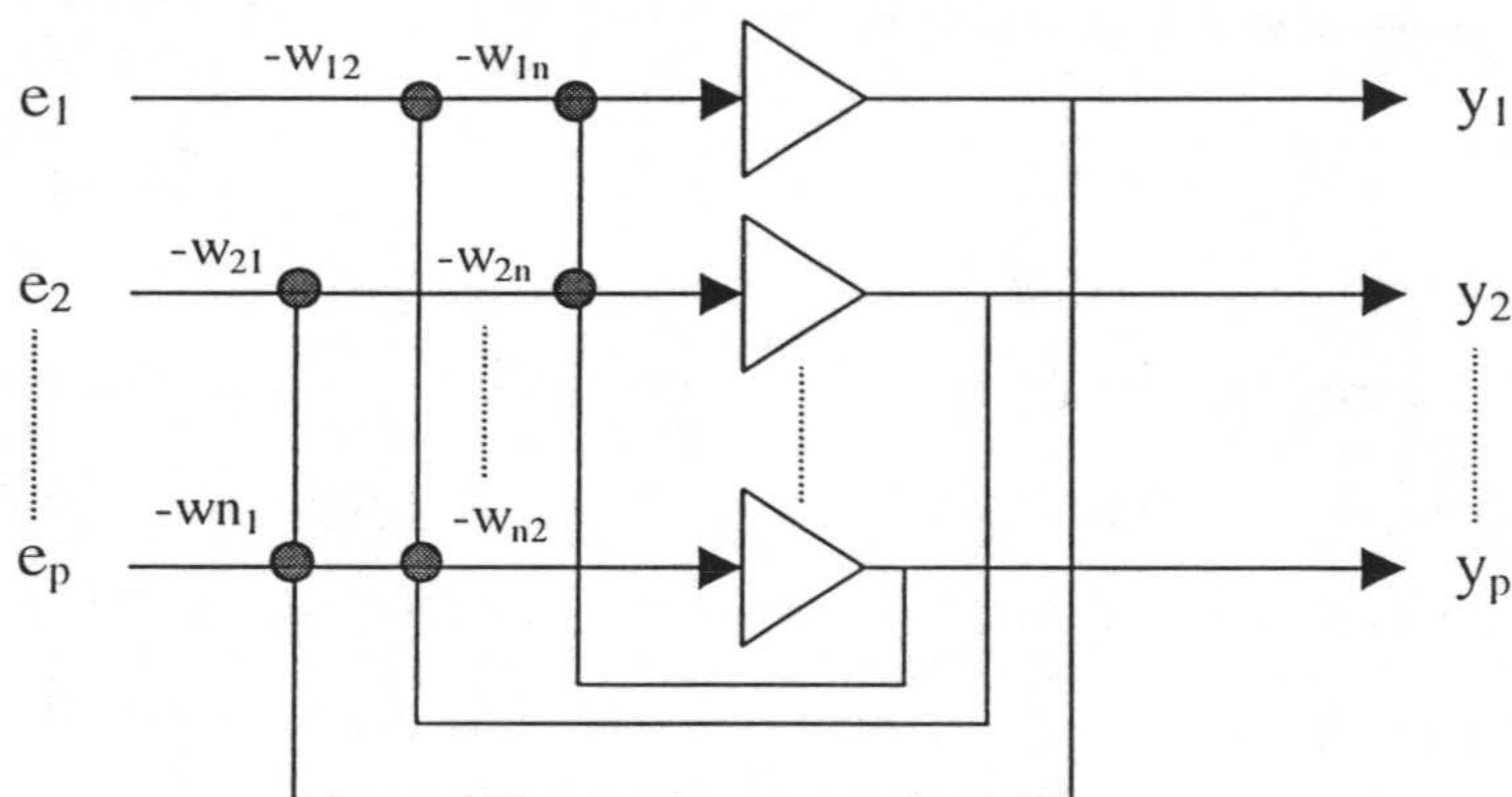


Figura 1.8. Arquitectura de la red neuronal Héault-Jutten.

Si las operaciones realizadas son muy rápidas respecto a las variaciones de las señales $e(t)$, y la red es estable, se puede hacer la siguiente aproximación:

$$\mathbf{y}(t) = (\mathbf{I} + \mathbf{W})^{-1} \cdot \mathbf{e}(t) \quad (1.46)$$

La regla de adaptación de pesos, \mathbf{W} , propuesta para garantizar la solución buscada es:

$$\frac{dw_{ik}}{dt} = \eta \cdot f[y_i(t)] \cdot g[y_k(t)] \quad (1.47)$$

donde η es la ganancia de aprendizaje (positiva).

Este algoritmo se puede expresar de forma recursiva como sigue:

$$w_{ik}(t+1) = w_{ik}(t) + \eta \cdot f[y_i(t)] \cdot g[y_k(t)] \quad (1.48)$$

Las funciones f y g son dos funciones no lineales que se introducen para generar los momentos de orden superior, que se obtienen a partir de los desarrollos en serie de Taylor de dichas funciones, y así poder aplicar un criterio de independencia estadística (Sec.A1.2). Es decir, dadas f y g se puede escribir:

$$\begin{aligned} f(x) &= \sum_j f_{2j+1} x^{2j+1} \\ g(x) &= \sum_m g_{2m+1} x^{2m+1} \end{aligned} \quad (1.49)$$

Teniendo en cuenta la expresión (1.47), el algoritmo converge si se verifica que:

$$E\left\{\frac{dw_{ik}}{dt}\right\} = \eta \cdot \sum_j \sum_m f_{2j+1} g_{2m+1} E\{y_i^{2j+1}(t) \cdot y_k^{2m+1}(t)\} = 0 \quad (1.50)$$

es decir, cuando todos los momentos de orden superior $E\{y_i^{2j+1}(t) \cdot y_k^{2m+1}(t)\}$ sean nulos. Por tanto, todos los coeficientes w_{ij} son estables gracias a la independencia estadística entre y_i y y_k , es decir, son estables cuando las señales de salida sean mutuamente independientes.

Ejemplos de las funciones f y g propuestas por Héroult y Jutten son:

Para señales subgaussianas:

$$\begin{aligned} f(x) &= x^3 & y & & g(x) &= x \\ f(x) &= \arctg(x) & y & & g(x) &= \text{sign}(x) \end{aligned} \quad (1.51)$$

Para señales supragaussianas:

$$f(x) = x \quad y \quad g(x) = x^3 \quad (1.52)$$

que, como se puede ver, son funciones no lineales impares.

La regla de adaptación es adecuada si se satisface que:

- ✓ Las fuentes son mutuamente independientes,
- ✓ $f(x)$ y $g(x)$ son funciones de media nula e impares.
- ✓ La matriz de mezcla A es lineal y no singular.

Este método muestra que cuando se aplica implícitamente un criterio de independencia a través de la anulación de los momentos de orden superior se llega a un criterio de convergencia con la consiguiente obtención de la matriz de pesos y de la separación de las fuentes originales.

Varios autores [MAR95, CIC96, LAH94] han estudiado este método, llegando a las siguientes conclusiones:

- ✓ El algoritmo es robusto en el sentido de que normalmente se fuerza a la red a tener un comportamiento estable. Sin embargo, la red no asegura la estabilidad global ni temporal. La estabilidad depende de muchos factores como las condiciones iniciales, los coeficientes de mezcla, o el tipo de funciones de activación escogidas.
- ✓ El algoritmo no funciona bien con señales mal escaladas y/o si la matriz de mezcla A está mal condicionada (es decir, es casi singular).
- ✓ En una implementación digital se requeriría el cálculo de la matriz inversa $\hat{W} = [I + W]^{-1}$ en cada iteración.
- ✓ La inicialización del algoritmo no puede ser cualquiera, ya que puede haber puntos de escape que conducirían a la divergencia.
- ✓ No existen implementaciones que permitan la separación de más de dos señales, siendo el procedimiento no válido para este caso ($p > 2$).

Vittoz y Arregit [VIT89], y Cohen y Andreou [COH92] realizaron una implementación VLSI de esta red.

Cichocki ha propuesto varias mejoras al algoritmo básico de Héroult-Jutten donde también se actualizan los elementos de la diagonal principal (ver Sec.1.12).

1.7 MÉTODOS BASADOS EN LA MINIMIZACIÓN DE LA DEPENDENCIA ESTADÍSTICA (ICA).

Estos métodos se fundamentan en utilizar la propiedad de que y_1, \dots, y_p son estadísticamente independientes si y sólo si (ver Sec.A1.23):

$$f_{y_1, \dots, y_p}(y_1, \dots, y_p) = \prod_{i=1}^p \tilde{f}_{y_i}(y_i) \quad (1.53)$$

donde $\tilde{f}_{y_i}(y_i)$ es la función de densidad de probabilidad marginal de y_i y $f_{y_1, \dots, y_p}(y_1, \dots, y_p) = f_y(y)$ es la función de densidad de probabilidad conjunta de $y=(y_1, \dots, y_p)$.

Llamando por comodidad:

$$\tilde{f}_y(y) \doteq \prod_{i=1}^p \tilde{f}_{y_i}(y_i) \quad (1.54)$$

la condición (1.53) se puede lograr expresando el producto de las distribuciones marginales $\tilde{f}_{y_i}(y_i)$ y la función de distribución conjunta, $f_y(y)$ en función de las señales observadas, $e(t)$ y de los elementos de la matriz \mathbf{B} , e imponer la condición de que ambas distribuciones $\tilde{f}_y(\mathbf{y}, \mathbf{B})$ y $f_y(\mathbf{y}, \mathbf{B})$ sean lo más próximas posible.

Esta aproximación se puede medir con la divergencia de Kullback-Leibler (Apéndice 2, Sec.A2.6) entre las funciones de densidad de probabilidad $\tilde{f}_y(\mathbf{y}, \mathbf{B})$ y $f_y(\mathbf{y}, \mathbf{B})$ que teniendo en cuenta la expresión (A2.39) puede indicarse como sigue:

$$D(f_y \parallel \tilde{f}_y; \mathbf{B}) = -h(y) + \sum_{i=1}^p \tilde{h}(y_i) \quad (1.55)$$

donde $D(f_y \parallel \tilde{f}_y; \mathbf{B})$ representa la divergencia Kullback-Leibler entre dos funciones f_y y \tilde{f}_y expresada en función de $\mathbf{B}=(b_{ij})$; $h(y)$ es la entropía conjunta del vector aleatorio y y $\tilde{h}(y)$ es la entropía marginal del componente i -ésimo de y .

La divergencia Kullback-Leibler es una “forma estadística” de cuantificar lo semejantes que son dos distribuciones de probabilidad y resulta ser una *función de contraste u objetivo* (Sec.1.3.1) válida para lograr la separación de fuentes; es decir:

$$\Phi_{MI}[y] = D(f_y \parallel \tilde{f}_y) \quad (1.56)$$

La divergencia de Kullback-Leibler entre las distribuciones de las componentes de un vector, usualmente se denomina *información mutua* entre dichos componentes; es decir, (1.55) y (1.56) representan la información mutua entre las componentes de y .

De acuerdo con la propiedad PA2.14 (Sec.A2.6):

$$\Phi_{MI}[y] \geq 0 \quad (1.57)$$

donde la igualdad se producirá cuando f_y y \tilde{f}_y sean iguales; es decir, cuando las componentes de y sean independientes.

Como $y = \mathbf{B} \mathbf{e}$, la entropía diferencial de y (1.55) puede expresarse como:

$$h(y) = h(\mathbf{B} \cdot \mathbf{e}) = h(\mathbf{e}) + \log|\det(\mathbf{B})| \quad (1.58)$$

según la propiedad PA2.8 (Sec.A2.15) descrita en el Apéndice 2.

El otro término de la expresión (1.55) implica conocer la entropía marginal de cada una de las componentes y_i de y ; es decir, $\tilde{h}(y_i)$. Para determinar esta entropía es necesario conocer la distribución marginal de y_i , que a su vez requiere eliminar los efectos de todos los otros componentes de y sobre y_i aislado. Se puede obtener una aproximación de $\tilde{h}(y_i)$ en función de los momentos de orden superior de la variable aleatoria y_i . Para ello se ha propuesto utilizar uno de los dos siguientes desarrollos en serie, truncados adecuadamente (a partir de un cierto término):

- *Desarrollo de Edgeworth*, que tiene la siguiente forma [COM94,STU94]:

$$\begin{aligned} \frac{f_y(y_i)}{N(0,1)} = & 1 + \frac{\kappa_{i,3}}{3!} H_3(y_i) + \frac{\kappa_{i,4}}{4!} H_4(y_i) + \frac{10\kappa_{i,3}^2}{6!} H_6(y_i) + \frac{\kappa_{i,5}}{5!} H_5(y_i) + \\ & + \frac{35\kappa_{i,3}\kappa_{i,4}}{7!} H_7(y_i) + \frac{280\kappa_{i,3}^3}{9!} H_9(y_i) + \frac{\kappa_{i,6}}{6!} H_6(y_i) + \frac{56\kappa_{i,3}\kappa_{i,5}}{8!} H_8(y_i) + \\ & + \frac{35\kappa_{i,4}^2}{8!} H_8(y_i) + \frac{2100\kappa_{i,3}^2\kappa_{i,4}}{10!} H_{10}(y_i) + \frac{15400\kappa_{i,3}^4}{12!} H_{12}(y_i) + \dots \end{aligned} \quad (1.59)$$

donde $N(0,1)$ representa la función de densidad de probabilidad de una variable aleatoria gaussiana estandarizada ($\langle y \rangle = 0$, $\sigma = 1$, ver Apéndice 1), $\kappa_{i,k}$ denota el cumulante de orden k de la variable aleatoria y_i , una vez estandarizada, y H_i representa el polinomio hermítico de orden i .

- *Desarrollo de Gram-Charlier* [AMA96b], que conduce a la siguiente expresión aproximada para la entropía marginal:

$$\begin{aligned} \tilde{h}(y_i) \approx & \frac{1}{2} \log(2\pi e) - \frac{\kappa_{i,3}^2}{12} - \frac{\kappa_{i,4}^2}{48} - \frac{(\kappa_{i,6} + 10\kappa_{i,3}^2)^2}{1440} + \\ & + \frac{3}{8} \kappa_{i,3}^2 \kappa_{i,4} + \frac{\kappa_{i,3}^2 (\kappa_{i,6} + 10\kappa_{i,3}^2)}{24} + \frac{\kappa_{i,4}^2 (\kappa_{i,6} + 10\kappa_{i,3}^2)}{24} + \\ & + \frac{\kappa_{i,4}^2 (\kappa_{i,6} + 10\kappa_{i,3}^2)^2}{64} + \frac{\kappa_{i,4}^3}{16} + \frac{(\kappa_{i,6} + 10\kappa_{i,3}^2)^3}{432} \end{aligned} \quad (1.60)$$

Sustituyendo en (1.55) las expresiones (1.58) y (1.60) se obtiene la divergencia Kullback-Leibler en función de los cumulantes:

$$\begin{aligned} D(f \parallel \tilde{f}; \mathbf{B}) = & -h(\mathbf{e}) - \log|\det(\mathbf{B})| + \frac{P}{2} \log(2\pi e) - \\ & - \sum_{i=1}^m \left(\frac{\kappa_{i,3}^2}{12} - \frac{\kappa_{i,4}^2}{48} - \frac{(\kappa_{i,6} + 10\kappa_{i,3}^2)^2}{1440} - \frac{3}{8} \kappa_{i,3}^2 \kappa_{i,4} - \right. \\ & - \frac{\kappa_{i,3}^2 (\kappa_{i,6} + 10\kappa_{i,3}^2)}{24} - \frac{\kappa_{i,4}^2 (\kappa_{i,6} + 10\kappa_{i,3}^2)}{24} - \\ & \left. - \frac{\kappa_{i,4}^2 (\kappa_{i,6} + 10\kappa_{i,3}^2)^2}{64} - \frac{\kappa_{i,4}^3}{16} - \frac{(\kappa_{i,6} + 10\kappa_{i,3}^2)^3}{432} \right) \end{aligned} \quad (1.61)$$

donde p es el número de observaciones, y los cumulantes (Sec.A1.2) son función de los elementos b_{ij} de \mathbf{B} .

Para obtener los elementos de \mathbf{B} se puede utilizar una red neuronal de pesos b_{ij} y éstos se calculan adaptativamente por el método del gradiente (Sec.1.4).

Para aplicar el método del gradiente se necesita en lugar de $D(f \parallel \tilde{f}; \mathbf{B})$, su derivada respecto a b_{ij} . En [HAY99] se demuestra que:

$$\frac{\partial}{\partial b_{ij}} D(f \parallel \tilde{f}; \mathbf{B}) = -(\mathbf{B}^{-T})_{ij} + \varphi(y_i) \cdot e_j \quad (1.62)$$

donde $\varphi(y_i)$ es una *función de activación* para el algoritmo de aprendizaje definida por:

$$\varphi(y_i) = \frac{1}{2} y_i^5 + \frac{2}{3} y_i^7 + \frac{15}{2} y_i^9 + \frac{2}{15} y_i^{11} - \frac{112}{3} y_i^{13} + 128 y_i^{15} - \frac{512}{3} y_i^{17} \quad (1.63)$$

El ajuste de pesos $\Delta \mathbf{B}$ aplicado en cada momento a la matriz de pesos \mathbf{B} se puede realizar aplicando el método del descenso de gradiente natural (Sec.1.4). Según este método:

$$\Delta \mathbf{B} = -\eta (\nabla D(f \parallel \tilde{f}(\mathbf{B}))) \mathbf{B}^T \mathbf{B} \quad (1.64)$$

El procedimiento anterior se aplica directamente a las señales observadas $e(t)$. Gaeta y Lacoume [GAE90] y Comon [COM94] desarrollaron un procedimiento similar, pero a señales previamente blanqueadas. Aplican el principio de máxima verosimilitud a la información mutua y realizan también un desarrollo de Gram-Charlier. Obtuvieron la siguiente función de contraste:

$$\Phi_{ICA}^0[y] = -\sum_{i=1}^p \kappa_{i,4}^2 + cte \quad (1.65)$$

donde $\kappa_{i,4}$ son los cumulantes de cuarto orden de las señales y_i .

1.8 ESTIMACIÓN DE MÁXIMA VEROSIMILITUD (ML).

La función de máxima verosimilitud es la función de densidad de probabilidad de un conjunto de datos en un modelo dado, pero vista como una función de los parámetros desconocidos del modelo.

La máxima verosimilitud es un procedimiento estadístico adecuadamente establecido para la estimación estadística. En esta sección primero se definirá una función logarítmica de máxima verosimilitud y después se optimizará con respecto al vector de parámetros del modelo probabilístico que se considere.

Sea $f_s(\bullet)$ la función de densidad de probabilidad de los vectores fuente s . Se puede probar [PAP84] que la función de densidad de probabilidad de las observaciones $e = A \cdot s$ se puede expresar así:

$$f_e(e, A) = \frac{f_s(A^{-1} \cdot e)}{|\det A|} \quad (1.66)$$

Sea $\mathcal{E} = \{e_m\}$, $m=1, M$ un conjunto de M vectores de mezcla independientes. A continuación se va a probar que una función de máxima verosimilitud adecuada es la siguiente:

$$f_e(\mathcal{E}, A) = \prod_{m=1}^M f_e(e_m, A) \quad (1.67)$$

Resulta más útil trabajar con una versión normalizada (dividida por el tamaño del conjunto M) del logaritmo de la función de verosimilitud (1.67):

$$\frac{1}{M} \log f_e(\mathcal{E}, A) = \frac{1}{M} \sum_{m=1}^M \log f_s(A^{-1} e_m) - \log |\det(A)| \quad (1.68)$$

Llamando $y = A^{-1} e$ a uno de los vectores de salida del bloque de recuperación, se tiene:

$$\frac{1}{N} \log f_c(\boldsymbol{\varepsilon}, \mathbf{A}) = \frac{1}{M} \sum_{m=1}^M \log f_s(\mathbf{y}_m) - \log |\det(\mathbf{A})| \quad (1.69)$$

Suponiendo $\mathbf{A}^{-1} = \mathbf{B}$ y denominando $f_y(\mathbf{y}, \mathbf{B})$ a la función de densidad de probabilidad de \mathbf{y} en función de los valores b_{ij} , haciendo tender $N \rightarrow \infty$ y considerando que el primer término del segundo miembro de (1.69) es la media del $\log f_s(\mathbf{y}_m)$, se tiene:

$$\begin{aligned} L(\mathbf{B}) &= E\{\log f_s(\mathbf{y})\} + \log |\det(\mathbf{B})| = \\ &= \int_{-\infty}^{\infty} f_y(\mathbf{y}, \mathbf{B}) \log f_s(\mathbf{y}) d\mathbf{y} + \log |\det(\mathbf{B})| \end{aligned} \quad (1.70)$$

donde $L(\mathbf{B})$ es la función de máxima verosimilitud logarítmica buscada. Se puede escribir:

$$f_s(\mathbf{y}) = \left(\frac{f_s(\mathbf{y})}{f_y(\mathbf{y}, \mathbf{B})} \right) \cdot f_y(\mathbf{y}, \mathbf{B}) \quad (1.71)$$

con lo que (A2.12), (A2.32):

$$\begin{aligned} L(\mathbf{B}) &= \int_{-\infty}^{\infty} f_y(\mathbf{y}, \mathbf{B}) \log \left(\frac{f_s(\mathbf{y})}{f_y(\mathbf{y}, \mathbf{B})} \right) d\mathbf{y} + \int_{-\infty}^{\infty} f_y(\mathbf{y}, \mathbf{B}) \log f_y(\mathbf{y}, \mathbf{B}) d\mathbf{y} + \log |\det(\mathbf{B})| = \\ &= -D(f_y \| f_s) - h(\mathbf{y}, \mathbf{B}) + \log |\det(\mathbf{B})| \end{aligned} \quad (1.72)$$

donde $h(\mathbf{y}, \mathbf{B})$ es la entropía diferencial del vector aleatorio \mathbf{y} en función de \mathbf{B} , y $D(f_y \| f_s)$ es la divergencia Kullback-Leibler entre $f_y(\mathbf{y}, \mathbf{B})$ y $f_s(\mathbf{y})$.

De (1.60) y (1.72) se obtiene [CAR98] que:

$$L(\mathbf{W}) = -D(f_y \| f_s) - h(\mathbf{e}) \quad (1.73)$$

donde $h(\mathbf{e})$ es la entropía diferencial de los vectores de mezcla. El único término del segundo miembro que depende de \mathbf{B} es la divergencia $D(f_y \| f_s)$, con lo que maximizar la función de máxima verosimilitud $L(\mathbf{B})$ es equivalente a minimizar la divergencia Kullback-Leibler, $D(f_y \| f_s)$; es decir, encajar o igualar al máximo las distribuciones de

probabilidad de las fuentes s con las señales que estiman las fuentes y ; objetivo que resulta evidente. En otras palabras, (1.67) es una función de máxima verosimilitud adecuada para estimar los valores (a_{ij}) , ya que maximizar (1.67) implica minimizar $D(f_y||f_s)$.

Se puede concluir, por tanto, que se puede asociar el principio de máxima verosimilitud con la siguiente función de contraste:

$$\Phi_{ML} = D(f_y||f_s) \quad (1.74)$$

Encontrar la matriz $\mathbf{B}=\mathbf{A}^{-1}$ maximizando (1.67) es lo mismo que encontrar la distribución de probabilidad $\mathbf{B}^{-1}\mathbf{e}$ que diverja lo menos posible de las funciones de distribución previsibles de las fuentes.

Relación entre el análisis de componentes independientes y el procedimiento de máxima verosimilitud.

En los métodos basados en la minimización de la dependencia estadística (ICA) se obtuvo como función de contraste (1.56): $\Phi_{MI}[y]=D(f_y||\tilde{f}_y)$.

Por otra parte, aplicando el principio de máxima verosimilitud se obtuvo (1.74): $\Phi_{ML}=D(f_y||f_s)$.

Las dos divergencias de Kullback-Leibler se pueden relacionar utilizando la descomposición pitagórica (Sec.A2.7) de la siguiente forma:

$$D(f_y||f_s) = D(f_y||f_{\tilde{y}}) + D(f_{\tilde{y}}||f_s) \quad (1.75)$$

En (1.75) el significado de los distintos términos es [AMA97], [CAR98]:

$D(f_y||f_s)$ es la divergencia entre las fuentes y las estimaciones de las mismas, y representa la *divergencia global* entre las fuentes estimadas y las fuentes originales.

$D(f_y \parallel f_{\tilde{y}})$ es la divergencia entre la función de distribución conjunta y el producto de las funciones de distribución marginales de las señales estimadas, y es una medida de la *divergencia estructural* que caracteriza el método de análisis de componentes independientes, y

$D(f_{\tilde{y}} \parallel f_s)$ es una medida de la *divergencia marginal* entre la distribución marginal de las señales producidas por el separador (y) y la distribución original del vector fuente (s).

Bajo la condición ideal de que $\mathbf{B}=\mathbf{A}^{-1}$, es decir, una separación de fuentes perfecta, las divergencias estructural y marginal se hacen cero. Es decir, la máxima verosimilitud y el análisis de componentes independientes conducen a la misma solución [CAR96], [AMA97], [HAY99].

1.9 MÉTODOS BASADOS EN EL PRINCIPIO DE MÁXIMA ENTROPÍA.

Bell y Sejnowski [BEL95a, BEL95b] propusieron en 1995 un *método de máxima entropía* para la separación ciega de fuentes. Demostraron que maximizar la entropía conjunta $H(\mathbf{x})$ de la salida de una red neuronal seguida de una no-linealidad, $x_i=g_i(y_i)$, (Figura 1.9) equivale a minimizar la información mutua entre las componentes y_i de salida de la red neuronal. La función $g_i(y_i)$ es una no-linealidad monótona, invertible, adecuadamente seleccionada, siendo $\mathbf{y}=\mathbf{W}\cdot\mathbf{e}$.

La no-linealidad tiene por objeto asegurar que la entropía diferencial, $h(\mathbf{x})$ sea acotada. El objetivo es realizar una estimación, $\mathbf{y}(t)$, del vector de fuentes originales $\mathbf{s}(t)$ maximizando la entropía $h(\mathbf{x})$ con respecto a \mathbf{W} . En definitiva, se trata de utilizar la siguiente función de contraste:

$$\Phi_{MI} = -h(\mathbf{G}(\mathbf{y})) \quad (1.76)$$

donde:

$$x_i = g_i(y_i) = \int_{-\infty}^{x_i} f(s_i) ds_i \quad x_i \in [0,1] \quad i = 1, \dots, p \quad (1.77)$$

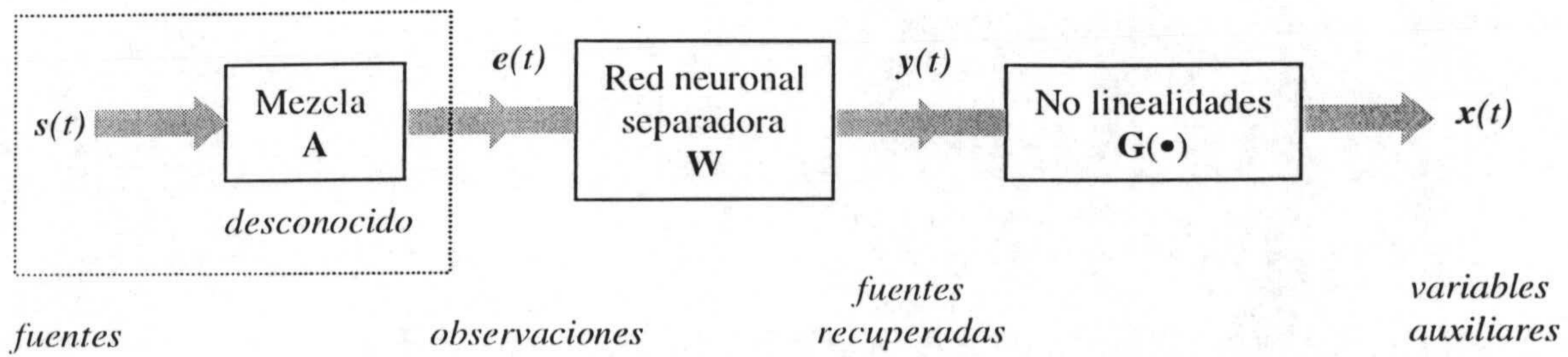


Figura 1.9. Método de máxima entropía para la separación de fuentes.

Minimizar Φ_{MI} , es decir, maximizar la entropía ($h[G(y)]$) del vector x a la salida de la no linealidad G es equivalente a hacer $W=A^{-1}$, lo que conduce a una perfecta separación de fuentes.

La no-linealidad G es la siguiente aplicación:

$$G : \begin{bmatrix} y_1 \\ \vdots \\ y_p \end{bmatrix} \rightarrow \begin{bmatrix} g_1(y_1) \\ \vdots \\ g_p(y_p) \end{bmatrix} = \begin{bmatrix} x_1 \\ \vdots \\ x_p \end{bmatrix} \quad (1.78)$$

que se puede expresar como (Figura 1.9):

$$x = G(y) = G(WAs) \quad (1.79)$$

donde, como la no-linealidad $G(\bullet)$ es invertible, es posible obtener el vector de fuentes originales, s , en función de la salida x :

$$s = A^{-1} W^{-1} G^{-1}(x) = \psi(x) \quad (1.80)$$

donde G^{-1} simboliza la siguiente aplicación:

$$G^{-1} : \begin{bmatrix} x_1 \\ \vdots \\ x_p \end{bmatrix} \rightarrow \begin{bmatrix} g_1^{-1}(x_1) \\ \vdots \\ g_p^{-1}(x_p) \end{bmatrix} = \begin{bmatrix} y_1 \\ \vdots \\ y_p \end{bmatrix} \quad (1.81)$$

La función de densidad de probabilidad del vector de salida x se puede expresar en función del vector de fuentes s de la siguiente forma [PAP84]:

$$f(\mathbf{x}) = \frac{f(\mathbf{s})}{|\det(\mathbf{J}(\mathbf{s}))|} \Big|_{\mathbf{s}=\psi(\mathbf{x})} \quad (1.82)$$

donde $\det(\mathbf{J}(\mathbf{s}))$ es el determinante de la matriz jacobiana $\mathbf{J}=(J_{ij})$, donde:

$$J_{ij} = \frac{\partial x_i}{\partial s_j} \quad (1.83)$$

Teniendo en cuenta (1.82) y (1.83) y A2.11, la entropía del vector aleatorio \mathbf{x} obtenida a la salida de la no-linealidad \mathbf{G} es:

$$\begin{aligned} h(\mathbf{x}) &= -E\{\log f(\mathbf{x})\} = -E\left\{\log\left(\frac{f(\mathbf{s})}{|\det(\mathbf{J}(\mathbf{s}))|}\right)\right\}_{\mathbf{s}=\psi(\mathbf{x})} = \\ &= -D(f(\mathbf{s}) \parallel |\det(\mathbf{J}(\mathbf{s}))|) \text{ evaluado en } \mathbf{s} = \psi(\mathbf{x}) \end{aligned} \quad (1.84)$$

En otras palabras, maximizar la entropía $h(\mathbf{x})$ es equivalente a minimizar la divergencia Kullback-Leibler entre $f(\mathbf{s})$ y una función de densidad de probabilidad definida por $|\det(\mathbf{J}(\mathbf{s}))|$ (ver Sec.A2.6).

Las funciones $g_i(y_i)$ se eligen de manera que tengan la forma de la función de densidad acumulativa, así las variables aleatorias x_i resultan ser acotadas en una intervalo $[0,1]$ y con distribuciones marginales uniformes. Por ejemplo, puede utilizarse una *función logística*:

$$x_i = \frac{1}{1 + e^{-y_i}} \stackrel{def}{=} g(y_i) \quad i = 1, \dots, p \quad (1.85)$$

En la Figura 1.10 puede comprobarse que esta función satisface los requisitos establecidos.

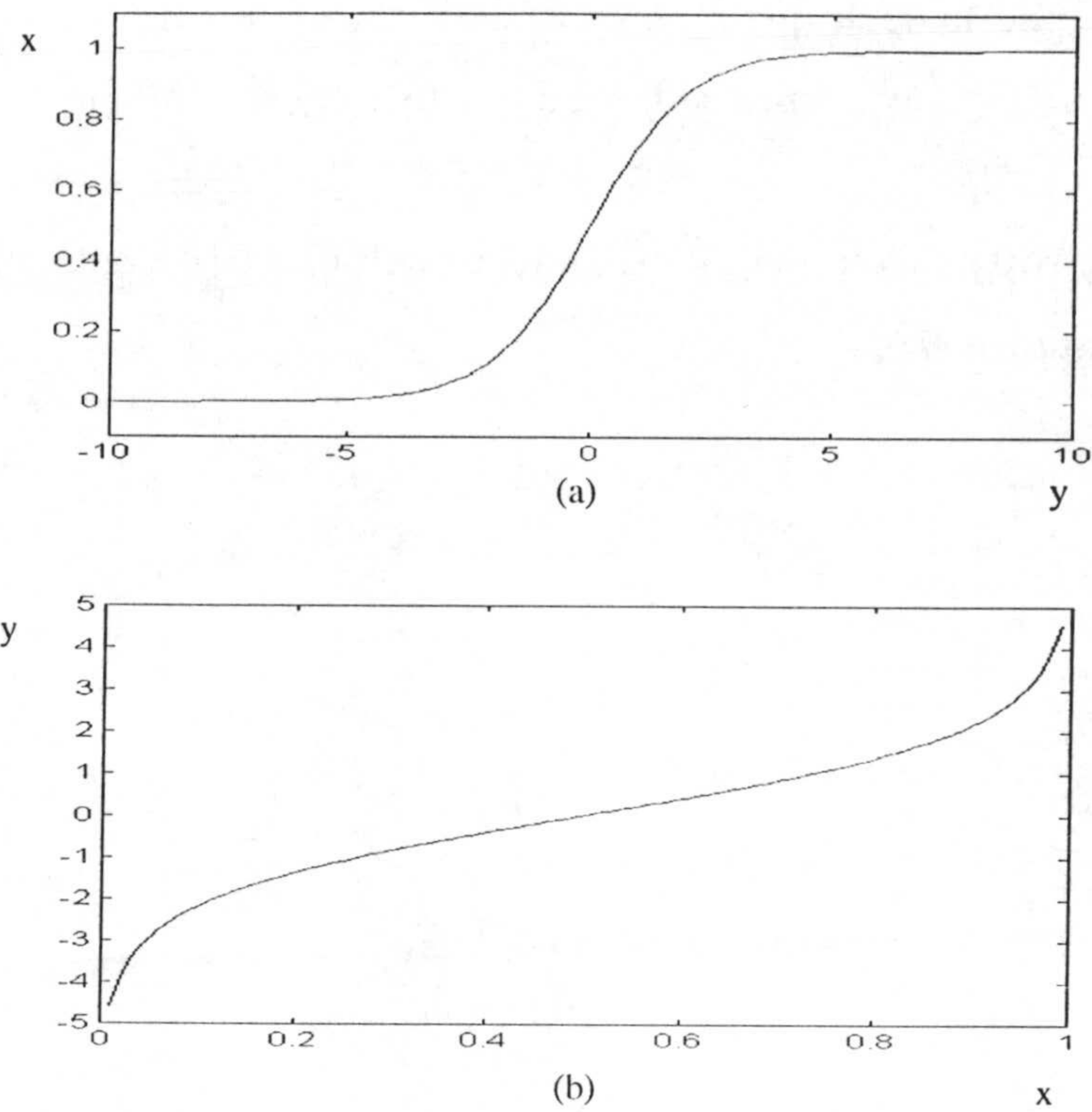


Figura 1.10. Función logística (a), y su inversa (b).

En la situación indicada x_i es una señal aleatoria *uniformemente distribuida* en el intervalo $[0, 1]$, y por tanto su entropía es $h_i(x_i) = \log|1-0| = 0$, (Apéndice 2, (A2.16)) y según (1.84):

$$f(s) = |\det \mathbf{J}(s)|_{s=\psi(x)} \quad (1.86)$$

Supóngase que la red neuronal está entrenada y por tanto se verifica $\mathbf{W} = \mathbf{A}^{-1}$; de (1.80) se tendrá:

$$s = \mathbf{G}^{-1}(x) = \psi(x) \quad (1.87)$$

Con lo que (1.86) se podrá expresar como:

$$f(s_i) = \left. \frac{\partial x_i}{\partial y_i} \right|_{x_i=g(s_i)} \quad i = 1, \dots, p \quad (1.88)$$

Recíprocamente se podría decir que si se satisface (1.88) la maximización de $h(\mathbf{x})$ conduce a $\mathbf{W} = \mathbf{A}^{-1}$ y la separación de señales se lleva a efecto como se pretendía.

Utilizando la regla de la cadena en cada elemento de la matriz jacobiana, (1.83) se puede expresar como (Figura 1.9):

$$J_{ij} = \frac{\partial x_i}{\partial s_j} = \sum_{k=1}^p \frac{\partial x_i}{\partial y_i} \cdot \frac{\partial y_i}{\partial e_k} \cdot \frac{\partial e_k}{\partial s_j} = \sum_{k=1}^p \frac{\partial x_i}{\partial y_i} \cdot w_{ik} \cdot a_{kj} \quad (1.89)$$

Es decir, la matriz jacobiana resulta ser:

$$\mathbf{J} = \mathbf{D} \cdot \mathbf{W} \cdot \mathbf{A} \quad (1.90)$$

donde \mathbf{D} es una matriz diagonal:

$$d_{ii} = \frac{\partial x_i}{\partial y_i}; \quad d_{ij} = 0 \quad \forall i \neq j \quad (1.91)$$

con lo que:

$$|\det(\mathbf{J})| = |\det(\mathbf{W} \cdot \mathbf{A})| \prod_{i=1}^p \frac{\partial g_i(y_i)}{\partial y_i} \quad (1.92)$$

De (1.84) se deduce que maximizar la entropía $h(\mathbf{G}(\mathbf{y}))$ conduce a minimizar (con respecto a \mathbf{W}) la esperanza del denominador $\log|\det(\mathbf{J})|$, ya que el numerador representa la distribución de las fuentes que se considera fija. Entonces, la función de contraste (1.76) resulta ser:

$$\Phi_{MI} = \log|\det(\mathbf{J})| \quad (1.93)$$

y teniendo en cuenta (1.92):

$$\Phi_{MI} = \log|\det(\mathbf{W})| + \sum_{i=1}^p \frac{\partial}{\partial W} \log\left(\frac{\partial x_i}{\partial y_i}\right) \quad (1.94)$$

El algoritmo de aprendizaje se puede obtener fácilmente a partir de (1.94) derivando con respecto a \mathbf{W} (Sec. 1.4):

$$\frac{\partial \Phi_{MI}}{\partial \mathbf{W}} = \mathbf{W}^{-T} + \sum_{i=1}^p \frac{\partial}{\partial \mathbf{W}} \log \left(\frac{\partial x_i}{\partial y_i} \right) \quad (1.95)$$

Utilizando la función logística (1.85) como no-linealidad, (1.95) se reduce a:

$$\frac{\partial \Phi_{MI}}{\partial \mathbf{W}} = \mathbf{W}^{-T} + (\mathbf{1} - 2\mathbf{x}) \cdot \mathbf{e}^T \quad (1.96)$$

donde \mathbf{e} es el vector de observaciones y $\mathbf{1}$ es un vector de unos.

Para obtener el algoritmo de aprendizaje (valores de modificación de pesos) se puede aplicar el método de *ascenso abrupto* ("steepest ascent") de forma que en cada iteración es cambio de pesos es:

$$\Delta \mathbf{W} = \eta \frac{\partial \Phi_{MI}}{\partial \mathbf{W}} = \eta [\mathbf{W}^{-T} + (\mathbf{1} - 2\mathbf{x}) \cdot \mathbf{e}^T] \quad (1.97)$$

Es más conveniente (según se indicó en la Sec.1.4) utilizar el gradiente natural, para lo cual basta según (1.24) con multiplicar el segundo miembro de (1.95) por $\mathbf{W} \cdot \mathbf{W}^T$. Operando se tiene que:

$$\mathbf{W}(m+1) = \mathbf{W}(m) + \mathbf{A} \cdot \mathbf{W}(m) = \mathbf{W}(m) + \eta (\mathbf{I} + (\mathbf{1} - 2\mathbf{x}(n))\mathbf{y}^T(n)) \cdot \mathbf{W}(n) \quad (1.98)$$

El algoritmo se aplica inicializando los elementos de \mathbf{W} , $w_{ij}(0)$, con valores muy pequeños y uniformemente distribuidos.

El algoritmo es válido sólo para distribuciones supra-gaussianas, pero Lee [LEE98] lo ha generalizado para poder separar también señales sub-gaussianas.

1.10 MÉTODOS BASADOS EN MEDIDAS ESTADÍSTICAS DE ALTO ORDEN.

Pueden usarse parámetros estadísticos de alto orden para definir funciones de contraste que resultan ser aproximaciones a las obtenidas a partir de los enfoques de máxima verosimilitud (ML). La información de alto orden se puede expresar sencillamente por medio de los cumulantes (ver Apéndice 1, Sec.A1.2.2).

Se puede demostrar, y en ello se basan diversos procedimientos [CAR89, 97, 98, 99, CHI93, LAC88], que un conjunto de señales (y_1, \dots, y_p) son estadísticamente independientes si y sólo si sus cumulantes cruzados se anulan (Sec.A1.2.3).

Como habitualmente se considera que las fuentes tienen funciones de densidad de probabilidad simétricas y están centradas en sus medias (es decir, $E\{y_i\}=0$) los momentos de orden impar son nulos (Sec.A1.2.2), por lo que es suficiente imponer las siguientes condiciones:

1. *Cumulantes cruzados de orden 2 nulos:* es decir, la matriz de correlación es diagonal (Sec.A1.2). Esta condición equivale a imponer $p(p+1)/2$ restricciones ya que la matriz de correlación es pxp y simétrica.
2. *Cumulantes cruzados de orden 4 nulos:* Con esta condición se completa la información necesaria para poder tener hasta pxp relaciones con objeto de calcular los b_{ij} de la matriz de reconstrucción **B**.

Es conveniente hacer notar que las señales con función de distribución gaussiana tienen todos los cumulantes de orden superior a 2 nulos, por lo que no se pueden separar este tipo de señales utilizando un enfoque estadístico (salvo en el caso de que tan sólo una de las fuentes sea gaussiana).

Los razonamientos que se hacen a continuación están fundamentados en conceptos básicos que se incluyen en el Apéndice 1.

En el caso de variables aleatorias con media cero, los cumulantes de 2º orden son idénticos a los momentos de 2º orden:

$$C_{12}[y] = Cum[y_1, y_2] \stackrel{def}{=} E\{y_1, y_2\} \quad (1.99)$$

y los cumulantes de 4° orden son:

$$C_{1234}[\mathbf{y}] = Cum[y_1, y_2, y_3, y_4] \stackrel{def}{=} E\{y_1, y_2, y_3, y_4\} - \\ - E\{y_1, y_2\}E\{y_3, y_4\} - E\{y_1, y_3\}E\{y_2, y_4\} - E\{y_1, y_4\}E\{y_2, y_3\} \quad (1.100)$$

Siempre que las variables aleatorias puedan dividirse en dos grupos que sean mutuamente independientes sus cumulantes cruzados serán cero. En consecuencia, la independencia más allá de la decorrelación de segundo orden puede ser comprobable (o imponible) utilizando los cumulantes de orden superior.

Como (por hipótesis, ver Sec.1.2) el vector de fuentes tiene sus componentes independientes, se tiene que:

$$C_{ij}[\mathbf{s}] = \sigma_i^2 \delta_{ij} \\ C_{ijkl}[\mathbf{s}] = \kappa_{i,4} \delta_{ijkl} \quad (1.101)$$

donde δ simboliza la delta de Kronecker, σ_i^2 la varianza, y $\kappa_{i,4}$ el cumunalnte de orden 4 de la señal s_i (ver Apéndice 1, Sec.A1.2):

$$\sigma_i^2 \equiv C_{ii}[\mathbf{s}] = E\{s_i^2\} \\ \kappa_{i,4} = C_{iiii}[\mathbf{s}] = E\{s_i^4\} - 3 \cdot E^2\{s_i^2\} \quad (1.102)$$

Puede obtenerse una medida de la divergencia entre la distribución de salida y la distribución del modelo de fuente, utilizando la diferencia cuadrática entre los cumulantes de entrada y salida (Figura 1.5e):

$$\begin{aligned}\Phi_2[\mathbf{y}] &\stackrel{def}{=} \sum_{i,j} (C_{ij}[\mathbf{y}] - C_{ij}[\mathbf{s}])^2 = \sum_{i,j} (C_{ij}[\mathbf{y}] - \sigma_i^2 \delta_{ij})^2 \\ \Phi_4[\mathbf{y}] &\stackrel{def}{=} \sum_{i,j,k,l} (C_{ijkl}[\mathbf{y}] - C_{ijkl}[\mathbf{s}])^2 = \sum_{i,j,k,l} (C_{ijkl}[\mathbf{y}] - \kappa_{i,4} \delta_{ij})^2\end{aligned}\tag{1.103}$$

Recuérdese (1.74) que la función de contraste $\Phi_{ML}[\mathbf{y}] = D(\mathbf{y}||\mathbf{s})$ también medía la diferencia (divergencia) entre la distribución de salida y la del modelo de fuentes.

Aisladamente $\Phi_2[\mathbf{y}]$ no es una función de contraste ya que $\Phi_2[\mathbf{y}] = 0$ sólo implica la decorrelación (no la independencia) entre las señales \mathbf{y} .

Caso de que las señales estuvieran previamente blanqueadas, es decir, $\mu_2 = 0$, $\sigma_i = 1$, y $\Phi_2[\mathbf{y}] = 0$, de (1.103) y (A1.13) se deduce que:

$$\Phi_4^0[\mathbf{y}] \stackrel{def}{=} -2 \sum_{i=1}^p \kappa_{i,4} C_{iiii}[\mathbf{y}] = E\{f_4(\mathbf{y})\}\tag{1.104}$$

En (1.104) se ha omitido un término aditivo constante ya que éste no influye en la obtención de la $\Phi_4^0[\mathbf{y}]$ original y se ha definido el factor:

$$f_4(\mathbf{y}) \stackrel{def}{=} -2 \sum_{i=1}^p \kappa_{i,4} (y_i^4 - 3)\tag{1.105}$$

En consecuencia, la función de contraste (1.105) es muy sencilla pues se reduce a obtener la expectación (valor medio) de una función de y_i .

Cardoso [CAR93] ha propuesto una función de contraste que utiliza un algoritmo desarrollado por él, denominado JADE, en el que utiliza un número menor de cumulantes cruzados y se aplica a señales blanqueadas. Esta función es la siguiente:

$$\Phi_{JADE}^0[\mathbf{y}] \stackrel{def}{=} \sum_{ijkl \neq ijkk} C_{ijkl}^2[\mathbf{y}]\tag{1.106}$$

donde la sumatoria se extiende a todas las combinaciones posibles de i, j, k, l , salvo aquéllas en las que aparezcan los dos últimos subíndices repetidos.

Teniendo cierta información acerca de las funciones de distribución de las fuentes se pueden obtener funciones de contraste más sencillas. Por ejemplo, si la kurtosis es negativa ($\kappa_{i,4} < 0, \forall i$) (función de distribución de probabilidad subgaussiana) de (1.106) y (1.105) se deduce que:

$$\min(E\{f_4(\mathbf{y})\}) = \min\left(E\left\{-2\sum_{i=1}^p -|\kappa_{i,4}|(y_i^4 - 3)\right\}\right) = \min E\left\{\sum_{i=1}^p y_i^4\right\} \quad (1.107)$$

con lo que resulta una nueva función de contraste:

$$\Phi_m^0[\mathbf{y}] = \sum_{i=1}^p E\{y_i^4\} \quad (1.108)$$

Por otra parte, si la kurtosis es positiva (función de distribución supragaussiana) la función de contraste resulta ser:

$$\Phi_m^0[\mathbf{y}] = -\sum_{i=1}^p E\{y_i^4\} \quad (1.109)$$

1.11 ANÁLISIS DE COMPONENTES PRINCIPALES NO LINEAL (PCA no lineal).

Oja, [OJA 91, 95, 97] propuso un método para la separación de fuentes basado en el análisis de componentes principales e implementable con una red neuronal (Figura 1.5e).

Considérese la siguiente regla de aprendizaje de tipo hebbiano para una de las neuronas i de la red neuronal de la Figura 1.11:

$$w_i(m+1) = w_i(m) + \eta(m)[e(m) - w_i(m)y_i(m)]y_i(m) \quad (1.110)$$

donde $w_i(m)$ es el vector de pesos en el instante m de la neurona con salida:

$$y_i(m) = w_i^T(m) \cdot e(m) \quad (1.111)$$

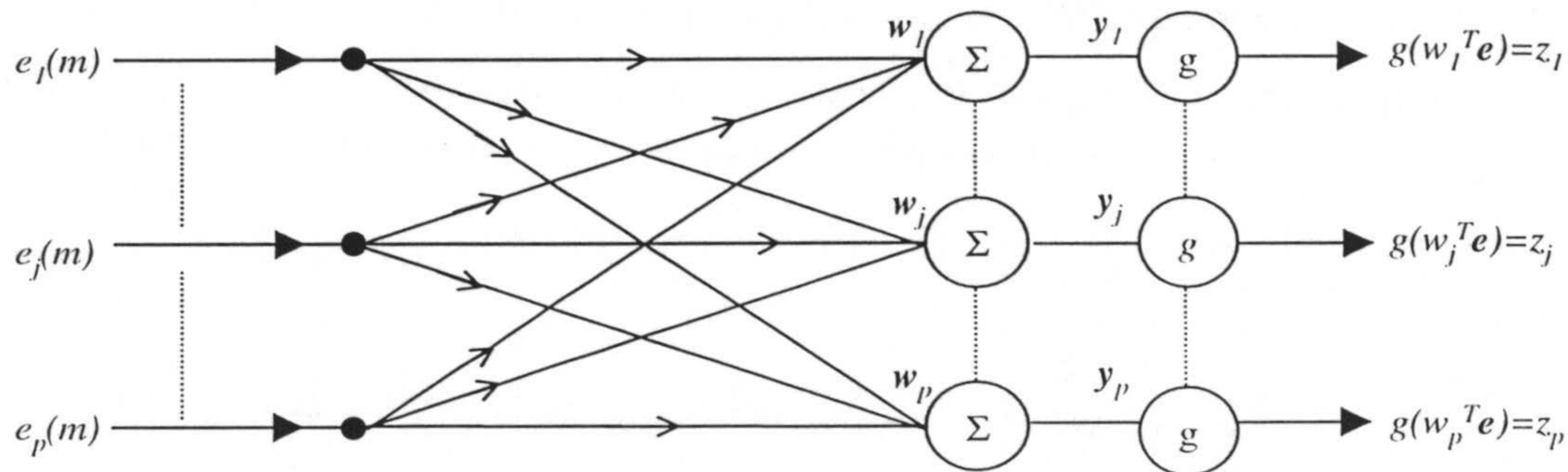


Figura 1.11. Red PCA no lineal.

Puede demostrarse [OJA85] que $w_i(m)$ tiende al vector propio dominante de la matriz de entrada $E\{e \cdot e^T\}$.

La expresión (1.110) puede ampliarse a las p neuronas de salida de la red neuronal:

$$W(m+1) = W(m) + \eta(m)[e(m) - W(m)y(m)]y(m) \quad (1.112)$$

donde ahora:

$$W(m) = [w_1(m), \dots, w_i(m), \dots, w_p(m)] \quad (1.113)$$

es la matriz de pesos cuyas columnas son los vectores de pesos individuales de cada neurona, y además:

$$y(m) = W^T(m) \cdot e(m) \quad (1.114)$$

Ahora los vectores pesos $w_i(m)$ son ortonormales y tienden al subespacio p -dimensional del espacio de vectores propios de la matriz de correlación de entrada.

Obviamente pueden ampliarse fácilmente las mismas reglas de aprendizaje a neuronas no lineales [OJA91]. Así, podríamos sustituir la expresión (1.111) que proporciona la salida de una neurona, y_i , en función de su vector de pesos w_i , y de la entrada en ese momento, e , por la siguiente (Figura 1.11):

$$g(y_i(m)) = g(\mathbf{w}_i^T(m) \cdot \mathbf{e}(m)) \quad (1.115)$$

siendo $g(\bullet)$ la función de activación de las neuronas (una función sigmoideal, por ejemplo). De esta forma, la regla de aprendizaje sería:

$$\mathbf{W}(m+1) = \mathbf{W}(m) + \eta(m) [\mathbf{e}(m) - \mathbf{W}(m)g(\mathbf{y}(m))]g(\mathbf{y}^T(m)) \quad (1.116)$$

siendo:

$$\mathbf{y}(m) = \mathbf{W}^T(m) \cdot \mathbf{e}(m) \quad (1.117)$$

Ahora el vector de salida será:

$$\mathbf{z} = g(y_i(m)) = [g(\mathbf{w}_1(m) \cdot \mathbf{e}(m)), \dots, g(\mathbf{w}_p(m) \cdot \mathbf{e}(m))] \quad (1.118)$$

La regla de aprendizaje (1.116) tiene un término $(\mathbf{e} \cdot g(\mathbf{y}^T))$ hebbiano (ya que es un producto de entrada por salida) y otro término $(g(\mathbf{y}) \cdot g(\mathbf{y}^T))$ normalizador que previene un crecimiento excesivo de los pesos.

Se ha probado [OJA97] que esta red funciona adecuadamente para separar imágenes y señales de voz. Si las densidades de probabilidad son subgaussianas (suaves) la función de activación debe ser sigmoideal y si son supragaussianas (abruptas) deben ser polinomiales (por ejemplo $g(u) = u^3$).

Karhunen [KAR94, 95a, 95b, 95c] ha demostrado que la regla de aprendizaje (1.116) puede obtenerse minimizando la función de contraste formada con la media del error cuadrático medio entre las observaciones \mathbf{e} y una estimación de éstas obtenidas como (ver Figura 1.5e):

$$\tilde{e} = W \cdot z \quad (1.119)$$

Es decir, dicha función de contraste es:

$$\Phi_{PCA_{nl}}[\mathbf{e}] \equiv E\left\{\|\mathbf{e} - \mathbf{W} \cdot \mathbf{z}\|^2\right\} = E\left\{\|\mathbf{e} - \mathbf{W} \cdot g(\mathbf{W}^T \cdot \mathbf{x})\|^2\right\} \quad (1.120)$$

1.12 OTROS ALGORITMOS.

Existen otros algoritmos de separación de fuentes, aunque la mayoría de ellos se basan en las ideas expuestas hasta ahora, y siguen los principios y criterios indicados en las Secciones 1.3 y 1.4. A continuación se describen resumidamente varias redes propuestas por Cichocki y Amari.

1.12.1 Red neuronal recursiva: Cichocki.

Cichocki y Amari [CIC95, 96] proponen unos algoritmos de aprendizaje no supervisado, robustos y eficientes que son válidos para problemas mal escalados o mal condicionados, basados en el principio usado por Héroult-Jutten (Sec.1.6). En los trabajos de Cichocki se propone una arquitectura hacia delante (“feedforward”) y se indica cómo obtener su arquitectura realimentada equivalente. Estos algoritmos de aprendizaje no supervisado, adaptativos, disminuyen o eliminan las desventajas del algoritmo propuesto por Héroult-Jutten. El principio es el mismo que el usado por Héroult-Jutten, es decir, conseguir la convergencia de los pesos a partir de la anulación de los momentos de orden superior para lograr la independencia estadística de las señales.

Algunos de los inconvenientes del modelo propuesto por Héroult-Jutten se pueden eliminar con la red neuronal cuyo esquema general se presenta en la Figura 1.12, que corresponde al modelo de la Figura 1.7 .

Considerando el mismo modelo de mezcla, la salida de la red formada por n neuronas lineales se puede expresar:

$$y_i(t) = \sum_{j=1}^p \hat{w}_{ij}(t) \cdot e_j(t) \quad i = 1, 2, \dots, p \quad (1.121)$$

o en forma matricial:

$$\mathbf{y}(t) = \hat{\mathbf{W}}(t) \cdot \mathbf{e}(t) \quad (1.122)$$

donde: $\hat{\mathbf{W}}(t) = [\hat{w}_{ij}(t)] \in \mathfrak{R}^{p \times p}$ es la matriz de pesos,

$\mathbf{e}(t) = [e_1(t), e_2(t), \dots, e_p(t)]^T$ es el vector de observación,

$\mathbf{y}(t) = [y_1(t), y_2(t), \dots, y_p(t)]^T$ es el vector de salida tras el aprendizaje.

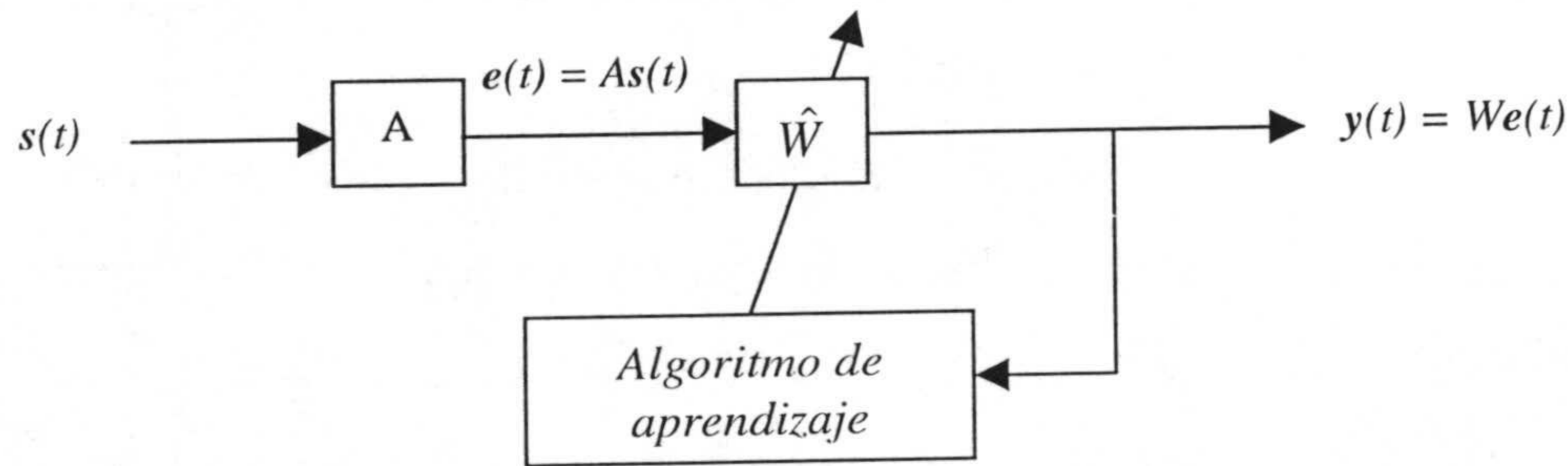


Figura 1.12. Diagrama de bloques del algoritmo hacia adelante.

La regla de adaptación propuesta es la siguiente:

$$\frac{d\hat{w}_{ij}}{dt} = \eta(t) \left[\lambda_i \cdot \hat{w}_{ij}(t) - f_i[y_i(t)] \cdot \sum_{k=1}^p \hat{w}_{kj}(t) \cdot g_k[y_k(t)] \right] \quad (1.123)$$

donde: $\eta(t)$ es la ganancia de aprendizaje que se puede fijar a priori,

$\lambda_i > 0 \forall i$, son factores de escala de la amplitud (normalmente $\lambda_i = 1 \forall i$),

f y g son dos funciones de activación.

En notación vectorial se tendría:

$$\frac{d\hat{\mathbf{W}}}{dt} = \eta(t) \{ \Lambda - \mathbf{f}[\mathbf{y}(t)] \cdot \mathbf{g}^T[\mathbf{y}(t)] \} \cdot \hat{\mathbf{W}} \quad (1.124)$$

con: $W(0) \neq 0$ y $\det W(0) \neq 0$ (usualmente: $W(0) = I$)

$$\Lambda = \text{diag}\{\lambda_1, \lambda_2, \dots, \lambda_n\} \quad (\text{usualmente : } \Lambda = I)$$

Este algoritmo es ligeramente más complejo que el de Héroult-Jutten, pero es más potente y tiene mejores prestaciones. El algoritmo se puede convertir directamente en una regla de adaptación discreta en el tiempo:

$$\hat{W}(m+1) = \hat{W}(m) + \eta(m) \{ \Lambda - f[\mathbf{y}(m)] \cdot \mathbf{g}^T[\mathbf{y}(m)] \} \cdot \hat{W} \quad (1.125)$$

con: $W(0) \neq 0$ y $\det W(0) \neq 0$, $m = (0, 1, 2, \dots)$ es el paso de aprendizaje, $t = mT_s$ y T_s el periodo de muestreo.

Las funciones de activación f y g son las mismas o similares a las que usaban Héroult-Jutten. Para asegurar la convergencia hacia la solución buscada hay que tomar f y g de forma que verifiquen:

$$f_i(y_i) = \begin{cases} y_i^p \text{sign}(y_i) & \text{para } p \text{ par} \\ y_i^p & \text{para } p \text{ impar} \end{cases} \quad (1.126)$$

$$g_j(y_j) = \tanh(10y_j) \quad \forall i, j \quad (p = 1, 2, 3, 4, 5)$$

El algoritmo presenta las siguientes diferencias con el modelo Héroult-Jutten:

- Es estable, independientemente de las condiciones iniciales.
- No se requiere el cálculo de matrices inversas en cada iteración.
- Asegura la auto-normalización de las señales de salida $\mathbf{y}(t)$. Es decir, el algoritmo tiene un mecanismo de control de ganancia auto-adaptativo inherente debido a la auto-adaptatividad de los pesos \hat{w}_{ii} .
- Permite separar señales con rangos de amplitud (energía) extremadamente amplios.
- La matriz de mezclas puede estar muy mal condicionada, es decir, los sensores pueden tener parámetros de transmisión muy parecidos (las señales $e_i(t)$ pueden ser muy parecidas).

La matriz \hat{W} describe el comportamiento dinámico del sistema global de mezcla/recuperación, y es completamente independiente de los parámetros de mezcla y escalado.

El modelo de red “hacia adelante” anterior (Figura 1.12) es relativamente sencillo de implementar tanto en software como en tecnología VLSI. Es temporalmente estable e independiente de las condiciones iniciales. Sin embargo, una de las desventajas de este modelo es que en algunos casos, los pesos sinápticos, especialmente con matrices de mezcla mal condicionadas o mal escaladas, pueden tomar valores excesivamente grandes. Esto es un inconveniente si la red se realiza en hardware ya que algunos de sus bloques constitutivos se saturarían. Estos valores grandes también harían difícil una implementación analógica de la red. Estas desventajas se pueden eliminar usando la red neuronal recurrente de la Figura 1.13.

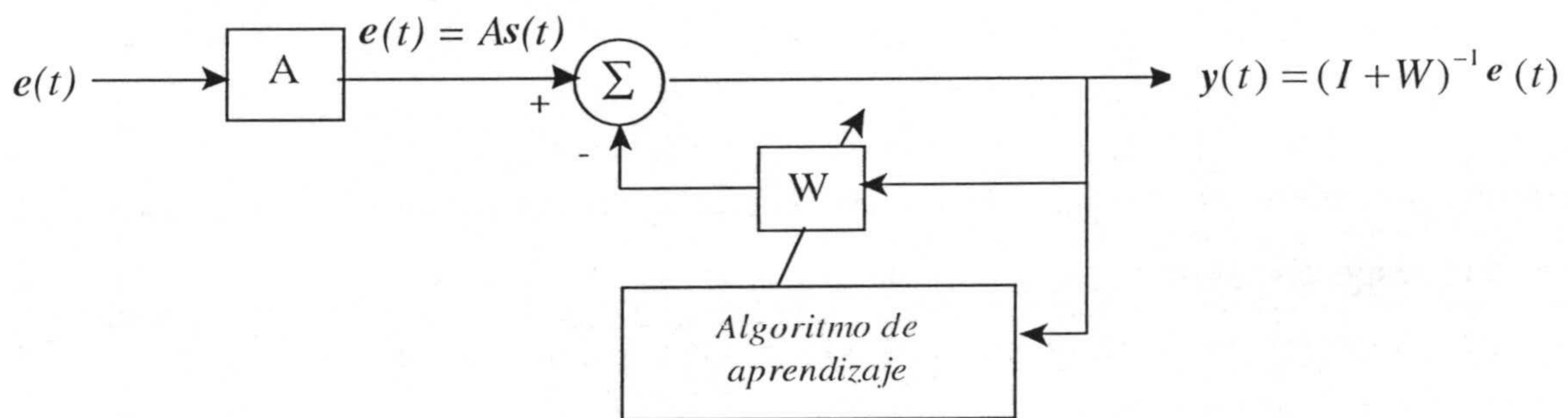


Figura 1.13. Esquema de red recurrente propuesto por Cichocki y Amari.

Las salidas de la red ahora son:

$$y_i(t) = e_i(t) - \sum_{j=1}^p w_{ij}(t) y_j(t) \quad (i = 1, 2, \dots, p) \quad (1.127)$$

o en forma matricial:

$$y(t) = e(t) - W(t)y(t) \quad (1.128)$$

reagrupando:

$$y(t) = [I + W(t)]^{-1} \cdot e(t) \quad (1.129)$$

Este modelo es completamente recurrente, es decir, hay conexiones entre cualquier par de unidades de proceso y también de cada unidad consigo misma ($w_{ii} \neq 0$). Tras el aprendizaje, la red neuronal propuesta satisface las siguientes expresiones:

$$y(t) = [I + W]^{-1} e(t) = [I + W]^{-1} A \cdot s(t) = D \cdot C \cdot s(t) \quad (1.130)$$

por tanto:

$$A = (I + W) \cdot D \cdot C \quad (1.131)$$

donde $D \in \mathcal{R}^{p \times p}$ es una matriz diagonal cuyos elementos son no nulos, y $P \in \mathcal{R}^{p \times p}$ es una matriz de permutación.

Comparando las expresiones (1.23) y (1.129), se observa que para transformar el algoritmo de aprendizaje propuesto para el modelo "hacia adelante" en otro equivalente para la red recurrente basta con hacer:

$$\hat{W}(t) = [I + W(t)]^{-1} \quad (1.132)$$

donde \hat{W} y W son las matrices de los pesos sinápticos de las redes "hacia adelante" y recurrente, respectivamente.

Teniendo en cuenta este último cambio, el algoritmo resultante es (1.124):

$$\frac{dW(t)}{dt} = -\eta(t) [I + W(t)] \left\{ \Lambda - f[y(t)] g^T [y(t)] \right\}$$

con: $W(0) \neq -I$ (normalmente $W(0) = 0$) (1.133)
 $\Lambda > 0$ (normalmente $\Lambda = I$)
 $\eta(t) > 0$

El algoritmo de aprendizaje adaptativo continuo se puede transformar fácilmente en un algoritmo discreto en el tiempo:

$$W(m+1) = W(m) - \eta(m) [I + W(m)] \left\{ \Lambda - f[y(m)] g^T [y(m)] \right\} \quad (1.134)$$

En este caso, hay que iniciar el algoritmo adecuadamente (normalmente $W(0)=0$) y la ganancia de aprendizaje debe ser suficientemente pequeña ya que de otra forma obtendríamos un comportamiento inestable.

Los algoritmos propuestos son en tiempo real ya que no necesitan almacenar temporalmente los datos de entrada (Sec.1.4). Además, son extremadamente robustos ya que llevan a cabo la separación de forma óptima incluso si la matriz de mezcla está mal condicionada.

1.12.2 Otras redes propuestas por Cichocki y Amari.

La red descrita en la Sec.1.12.1 puede entenderse de forma sencilla introduciendo la matriz $G(y)$ definida como:

$$G(y) \stackrel{def}{=} \Lambda - \mathbf{f}(y) \cdot \mathbf{g}^T(y) \quad (1.135)$$

con lo que la actualización de pesos (1.125) se puede expresar como:

$$\hat{W}(m+1) = \hat{W}(m) + \eta(m) \cdot G(y(m)) \quad (1.136)$$

Cichocki y Amari [AMA95a], [CIC94, 95, 96] también definieron otras redes en las que $G(y)$ tiene los siguientes valores:

$$G(y) = I - y \cdot y^T - g(y)h^T(y) + h(y)g^T(y) \quad (1.137)$$

$$G(y) = I - \sum_{i=0}^r y(m) \cdot y_{m-iT}^T \quad (1.138)$$

En (1.138) T representa un retardo de tiempo elegido adecuadamente.

Esta dos nuevas redes pueden entrenarse mediante la expresión (1.136).

Como de la red correspondiente a (1.135) se pueden obtener versiones de redes recurrentes utilizando el siguiente algoritmo de aprendizaje:

$$W(m+1) = W(m) - \eta(m)(W(m) + I) \cdot G(y(m)) \quad (1.139)$$

la salida (señales recuperadas) resulta ser:

$$y(m) = [I + W(m)]^{-1} \cdot x(m) \quad (1.140)$$

1.13 APLICACIONES DE LA SEPARACIÓN CIEGA DE FUENTES.

Siempre que se tenga un sistema que emita señales que se propaguen por un medio, se reciban en unos sensores y e interese recuperar dichas fuentes, se tiene un posible campo de aplicación de la separación ciega de fuentes. Entre este amplio margen de aplicaciones se pueden encontrar, por ejemplo, todas aquellas en las que interese eliminar un ruido, la biomedicina, sistemas de audio, radiocomunicaciones, etc. A continuación se presentan a grandes rasgos los campos de aplicación más usuales y que más se han desarrollado hasta el momento.

Aplicaciones derivadas de la supresión del ruido.

Uno de los problemas que mayor interés despierta en muchos campos del ámbito científico y tecnológico es el de la *supresión de ruido*: Como se ha comentado anteriormente (Sec.1.1) y se comentará en la Sec.1.14, la separación de señales permite la extracción de fuentes de ruido para disponer de señales útiles en diversas aplicaciones. Desde hace tiempo, se han estudiado y detectado numerosos tipos de ruido en diversos dispositivos [GUP77]; por ejemplo, el producido por efecto Shot (transporte), el generado por efecto Flicker (modulación), el ruido térmico (fluctuaciones), el ruido Johnson, el ruido impulsivo (atmosférico y solar) y el ruido cuántico (optoelectrónica, superconductores, dispositivos de electrones calientes). Por otra parte, las fuentes de ruido pueden ser *internas*, presentes en los conductores, semiconductores, radiorreceptores, diodos, amplificadores, osciladores, transistores, circuitos, TV y generadores de pulsos, entre otros, y *externas*, afectando a antenas, comunicaciones, receptores de UHF, scattering, sistemas no lineales,

comunicaciones de pulsos, sonar, transmisión de datos binarios, modulación PCM, radar y radionavegación, entre otros.

Un ejemplo concreto de la importancia de la eliminación de ruido en sistemas reales se tiene en la *voz en telecomunicaciones*, existiendo una demanda creciente de aplicaciones controladas por la voz que exhiban prestaciones robustas e independientes del medio.

Los sistemas con conjuntos de micrófonos están siendo utilizados en captura de la voz, localización de fuentes acústicas y en aplicaciones tales como teléfonos móviles, videoconferencias e interfases de audio con sistemas basados en PC [SIL97]. Los inconvenientes de estos sistemas son la dependencia con la geometría del conjunto de micrófonos, las características acústicas del medio y la degradación de las señales debida a la reverberación, así como las prestaciones no lineales de los sensores acústicos.

Aplicaciones biomédicas.

En relación con las *señales biomédicas*, la separación de fuentes es una técnica prometedora en este campo, dado que los métodos tradicionales de identificación y medida de respuestas se basaban en las amplitudes y latencias de los picos de la forma de onda y en modelos de dipolos múltiples que proporcionaban resultados ambiguos cuando la geometría de las fuentes es desconocida o compleja [SCO96]. De hecho, existen en la actualidad líneas de investigación para aplicaciones de tratamiento de señal en biomedicina relacionadas con el registro de señales [SER96] y con la captación de varias señales con multisensores. Además, en la captación de señales biomédicas se usan varios sensores situados próximos unos de otros, con el inconveniente de que no sólo se graban las señales deseadas sino también otras procedentes de otros fenómenos biológicos, lo que proporciona un ambiente idóneo para aplicar la separación de fuentes. Por ejemplo, Lee [LEE98] presenta un método basado en ICA para separar, a partir de encefalogramas, señales de distinta naturaleza como pueden ser el movimiento de los ojos, ruido producido por los músculos, o el latido del corazón. El análisis de imágenes obtenidas por resonancia magnética permite controlar a una persona mientras realiza tareas psicomotrices, sin embargo estas lecturas suelen ir contaminadas por otras señales lo que impide un buen seguimiento de las que interesan. En este campo también se ha aplicado ICA para buscar una solución [LEE98]. Dentro del mismo campo se encuentra la reconstrucción ciega de flujo autónomo cardiaco en el rango de bajas

frecuencias (LF) usando solo el ritmo cardiaco y la presión arterial de la sangre [VET99]. Otras aplicaciones de interés son la descomposición de potenciales evocados obtenidos con un magnetoencefalógrafo (MEG) producidos después de una estimulación sensorial [VIG99], la separación de patrones de señales espaciales a partir de grabaciones de imágenes ópticas [SCH99], separación de señales procedentes de un electrocardiograma del corazón de una mujer embarazada y su feto [LAC97], etc.

Aplicaciones de audio.

Uno de los orígenes de la separación de fuentes se basó en encontrar una explicación al efecto “coctail party”. Este efecto consiste en la habilidad del ser humano de poder centrar su atención y escuchar a una sola persona, aún habiendo más personas hablando a la vez o existiendo otras fuentes sonoras. Por tanto, una de las primeras aplicaciones que surgieron fue la de separar señales de audio, voz por ejemplo, en distintos entornos. Algunas referencias interesantes sobre el tema son:

- Separación de señales en mezclas de audio y voz [BOZ96], [PRI99a], [ROD96], [TOR99], [WES99].
- Separación de señales reverberantes grabadas con varios micrófonos [WU99].
- Separación de señales usando un conjunto compacto de micrófonos en un entorno reverberante [NGO99].
- Control de un entorno en el que funcionan simultáneamente varias máquinas [LAC97].

Aplicaciones en radiocomunicaciones.

Otro campo interesante y de gran actualidad es el de las radiocomunicaciones, donde se pueden encontrar las siguientes referencias:

- Tratamiento de antenas [CHE96]. Por ejemplo, se parte de las señales captadas por una serie de antenas producida al incidir varias señales de banda estrecha originadas por fuentes situadas en lugares desconocidos. De nuevo lo que interesa es obtener aisladas dichas fuentes.
- Radar [CHA96], [LAC97].
- Comunicaciones móviles [FEN99].
- Separación de mensajes en sistemas de comunicación de múltiple acceso con división de código (CDMA). [RIS99].

- Enlaces ionosféricos de alta frecuencia (HF) [CHE99].
- Señales de identificación por radiofrecuencia de blancos múltiples utilizando separación de señales con redes neuronales artificiales. [DEV95, 96a, 96b, 99b].
- Satélite de telecomunicaciones [BOZ96].
- Análisis hiperespectral con ICA [BAY].

Otras aplicaciones.

Existen otros campos de aplicación. A continuación se enumeran algunos ejemplos:

- Separación de dos fuentes ópticas infrarrojas [KOP99].
- Eliminación del ruido en imágenes [HYV99] y procesamiento, en general, de imágenes [CIC95], [SAH96], [PUG99], [BEL97].
- Vigilancia de instalaciones EDF [D'UR96].
- Control de máquinas eléctricas rotativas. En una sala con gran cantidad de máquinas eléctricas se puede detectar el sonido individual producido por cada una de ellas, y localizar si se produce alguna avería en alguna máquina. [CAP96], [YPM99], [GEL99].
- Análisis de datos sísmicos [THI96]
- Identificación de los componentes de una mezcla de sustancias químicas a partir de una espectroscopia de Resonancia Magnética Nuclear [NUZ99]
- Clasificación no supervisada [LEE98].
- Sensores inteligentes [PAR98]. En sensores eléctricos y magnéticos la señal a detectar suele estar enmascarada con señales de otro origen, usualmente térmico. Se trata de separar automáticamente la influencia de las distintas fuentes.
- Análisis de datos financieros [BAC98].

1.14 AMPLIACIONES DEL MODELO BÁSICO. LÍNEAS ABIERTAS.

Hasta ahora en este capítulo se ha realizado una revisión acerca de los trabajos más significativos que tratan de resolver el problema de separación ciega de fuentes; pero únicamente se ha considerado fundamentalmente el modelo básico lineal de mezclas instantáneas descrito por la expresión (1.10).

A continuación se enumeran y citan aportaciones relevantes que siguen los modelos más generales planteados en la Sec.1.1, o que no necesitan cumplir todas las hipótesis indicadas en la Sec.1.2.

1.14.1 Datos y mezclas no estacionarios.

Lo usual es suponer que los elementos a_{ij} de la matriz de mezcla y las fuentes s_i son estacionarias. En realidad puede ocurrir que el medio varíe con el tiempo, con lo que los elementos de la matriz \mathbf{A} , que lo modelan, cambien también. Por este motivo, es importante en la práctica que los algoritmos de separación puedan adaptarse a cambios (lentos) de los coeficientes de la mezcla, e incluso de la distribución estadística de las fuentes.

Dadas las peculiaridades de las redes neuronales artificiales [PRI96, HAY99], la utilización de éstas para implementar sistemas no estacionarios de separación de fuentes puede resultar muy eficiente. En efecto, es posible conseguir que los elementos de matriz de recuperación se auto-adapten en el tiempo a los cambios de la matriz de mezcla, aplicando constantemente la regla de aprendizaje seleccionando adecuadamente el parámetro de ganancia de aprendizaje $\eta(t)$ en el tiempo.

En [MAT95] se propone una red neuronal que sólo considera la no estacionalidad (en realidad una cicloestacionalidad) de las fuentes. En [NGU95] se muestran algunos resultados experimentales para datos no estacionarios de voz.

Por otra parte, Cichocki en [CIC96] propone un algoritmo auto-adaptativo para entornos no estacionarios, basado en la adecuada modificación de la ganancia de aprendizaje a lo largo del tiempo.

1.14.2 Efecto del ruido.

La mayoría de los autores consideran el ruido como una fuente más desconocida, o alternativamente, el término $n(t)$ del modelo (1.10) igual a cero. Según Karhunen [KAR96a] el primer enfoque no es correcto ya que un modelo de ruido más preciso

implicaría además de añadir una componente (fuente) independiente de las demás señales, habría que considerar otras componentes superpuestas a las fuentes originales.

En cualquier caso se observa que el ruido empeora notablemente las prestaciones de los algoritmos de separación (ver por ejemplo las Sec.4.7 y 5.6 de la presente memoria).

Si el número de observaciones q es mayor que el número de muestras p es posible filtrar el ruido utilizando, por ejemplo, un blanqueo PCA (Sec.1.5) que automáticamente reducirá las q componentes iniciales a las p más adecuadas para la separación.

En relación con el ruido, Cardoso [CAR98] opina que es una cuestión abierta determinar los dominios de aplicación donde realmente es de interés considerar modelos de ruido.

1.14.3 Estimación del número de fuentes, p .

En aplicaciones reales, el número de fuentes puede ser desconocido e incluso variable en el tiempo. No obstante, la mayoría de los algoritmos de separación de señales son particularmente eficientes si el número de sensores es igual al número de fuentes, y parten de hecho de esta hipótesis. Caso de que no se verifique $p=q$ hay dos posibilidades:

- a) *Más sensores que fuentes, $q > p$:* En este caso puede usarse una etapa previa de blanqueo que transformará las q observaciones en las p señales decorrelacionadas con mejor relación señal/ruido. Como se vio en la Sec.1.5 esto se puede realizar con un PCA. Primero se estima la matriz de covarianza de las observaciones $E\{y(m)y^T(m)\}$ y después se calculan sus valores propios. Las p señales proyectadas sobre los vectores propios correspondientes a los p valores propios mayores resultan ser las más adecuadas (las $q-p$ señales resultantes pueden considerarse ruido). Como se indicó en la Sec.1.5 este análisis puede realizarse con una red neuronal artificial (ver HAY99, por ejemplo).
- b) *Menos sensores que fuentes, $q < p$:* El problema resulta irresoluble ya que las q observaciones no contienen información suficiente para separar exactamente las p fuentes. Se pueden estimar las p señales más potentes, pero con una distorsión provocada por las otras $p-q$ señales. Zhu estudió teóricamente este problema consiguiendo separar las p fuentes en q grupos distintos en los que se obtuvieron las

fuentes casi puras o combinaciones lineales de ellas [ZHU95]. Se pueden encontrar algunos resultados experimentales con el algoritmo PCA- no lineal (Sec.1.11) en [KAR97]. La situación descrita, es decir, la descomposición de un vector aleatorio en una suma de componentes estadísticamente independientes, pero no necesariamente cada una de ellas monodimensional, podría denominarse *Análisis multidimensional de componentes independientes (MICA)* [CAR98].

1.14.4 Mezclas mal condicionadas.

Esta situación se produce cuando la matriz de mezcla es casi singular, es decir, cuando las mezclas son muy similares. El problema se soluciona diseñando o situando los sensores con una buena diversidad espacial, esto es, eligiendo varias geometrías u orientaciones y situándolos alejados unos de otros, cuando esto es posible.

Se han desarrollado algoritmos de separación de fuentes *equivariantes* de forma que las prestaciones no se ven influenciadas por los valores concretos de la matriz de mezcla [CAR96, AMA95a]. Esta es una cualidad muy buscada en la resolución del problema de separación de señales.

1.14.5 Modelos de mezcla más complejos que el lineal instantáneo.

Diversos autores ha abordado los distintos modelos citados en la Sec.1.1: el modelo no lineal [ALM99], [PUN99], el modelo lineal convolutivo [NGU95] y el post-no-lineal [TAL96,98], [PUN98a].

La separación de mezclas convolutivas tiene un gran interés en el campo de la audio-frecuencia y resulta mucho más difícil que el de mezclas instantáneas. Recuérdese (Sec.1.1) que en este caso considerado cada elemento de la matriz A se convierte en un filtro (Figura 1.2). El problema resulta complejo (y de gran interés) incluso en el caso de *identificación entrada/salida* (no ciega) debido al gran número de parámetros necesarios para describir los canales de audio.

1.14.6 Uso de información a priori.

Cualquier información sobre las fuentes puede dar lugar a simplificaciones en el algoritmo de separación o a mejorar notablemente los resultados. En particular, si la mezcla realiza únicamente una correlación de las fuentes, siendo estas gaussianas, se pueden separar sencillamente con una etapa de blanqueo.

Si las distribuciones de probabilidad de las fuentes son conocidas (señales de voz o música, por ejemplo) se puede simplificar el problema ([BEL95a, 95b], [PRI99]).

Si las señales son discretas (binarias o multivaluadas) se pueden utilizar algoritmos muy eficientes [PUN95a, 95b, 95c].

1.15 CONCLUSIONES.

En este capítulo se ha presentado una introducción general al problema de la separación ciega de fuentes, efectuando una revisión de los principales métodos propuestos hasta la fecha.

En primer lugar se ha realizado una descripción y planteamiento general del problema (Sec.1.1), y se han justificado las indeterminaciones e hipótesis que habitualmente se realizan en los algoritmos de separación (Sec.1.2). Estas dos primeras secciones sirven también para introducir la notación que se utilizará a lo largo de la presente memoria.

En vez de presentar uno a uno (aisladamente) los distintos procedimientos propuestos para la reconstrucción de señales mezcladas, se ha tratado de hacer abstracción tanto de los principios en que se basan, como de las arquitecturas que los implementan (Sec.1.3 y 1.4), buscando una unidad conceptual a partir de la cual se entiendan mejor los diversos métodos de separación. Una vez establecido este marco general y coherente, se resumen los métodos más relevantes publicados hasta el momento, que están basados en la minimización de la dependencia estadística (ICA, Sec.1.7), estimación de máxima verosimilitud (ML, Sec.1.8), principio de máxima entropía (ME, Sec.1.9), medidas

estadísticas de alto orden (Sec.1.10) y el análisis de componentes principales no lineal (PCA no lineal, Sec.1.11).

Desde que en Abril de 1986 Jeanny Héroult y Christian Jutten [Her86], en una reunión sobre Computación con Redes Neuronales celebrada en Snowbird (Utah), enunciaron por primera vez el problema de separación de fuentes y presentaron un modelo de red neuronal recurrente con una regla de aprendizaje hebbiana capaz de separar mezclas lineales de señales independientes, no han dejado de surgir aplicaciones reales de este tipo de sistemas. La Sec.1.13 trata de enumerar algunas de las de mayor interés, encuadradas en campos tales como separación de señales de voz, procesamiento de señales captadas por grupos de antenas, registros biomédicos multisensoriales, análisis de datos financieros, etc.

El capítulo concluye (Secc.1.14) con la enumeración de diversas aportaciones que versan sobre modelos de mezcla más complejos que el lineal instantáneo, y de otros temas colaterales que perfilan distintas líneas de investigación abiertas en el campo de la separación ciega de fuentes.

CAPÍTULO 2:

FUNDAMENTO TEÓRICO DE LOS MÉTODOS GEOMÉTRICOS DE SEPARACIÓN DE FUENTES

En el Capítulo 1 (Sec.1.3) se indicó que el problema de separación ciega de fuentes se puede abordar bajo dos perspectivas: estadística y geométrica, y dicho capítulo se dedicó a analizar los fundamentos de los métodos ideados bajo la primera. En el presente capítulo se aborda el problema de la separación ciega de señales haciendo un análisis teórico del mismo desde un enfoque geométrico. Se comienza (Sec.2.1) analizando el modelo matemático de la mezcla de señales, que representa una mezcla lineal de las fuentes en el mismo medio en que se transmiten. También se modela la influencia sobre las mezclas de un ruido añadido a las mismas en el medio (Sec.2.1.2). Los métodos geométricos de separación se basan en propiedades geométricas y algebraicas de la transformación que existe entre las fuentes y las mezclas, y se fundamentan en una serie de hipótesis que se razonan en la Sec.2.2. Posteriormente (Sec.2.3) se muestran las propiedades en que se basa el método geométrico de separación de fuentes. A partir de dichas propiedades se deduce que para facilitar el desarrollo de los distintos métodos geométricos propuestos hasta

ahora es necesario obtener de forma sencilla los vértices del hiperparalelepípedo del espacio de observación, por lo que en la Sección 2.4. se describen diversas formas de hacerlo. Para poder medir la calidad de los resultados obtenidos con los algoritmos de separación, en la Sección 2.5 se presentan distintas métricas, como el índice de prestaciones de Amari [AMA95,96], con el que con un solo número se obtiene una medida de las prestaciones y otros índices basados en el error cuadrático medio entre la matriz de transformación lineal original y la matriz de separación obtenida, o en la diafonía que mide la desigualdad entre dos señales.

2.1. MODELO MATEMÁTICO DE LA MEZCLA LINEAL DE SEÑALES

Según se indicó en la Sec.1.1, el problema de la separación ciega de fuentes implica la obtención de las señales generadas por p fuentes, s_j , ($j = 1, \dots, p$), a partir de las mezclas, e_i , ($i=1, \dots, p$), detectadas por p sensores. La *mezcla de las señales* tiene lugar en el medio en el que se están propagando. Una de las situaciones de mayor interés es el de las *mezclas lineales*. En este caso, las señales observadas corresponden a las fuentes originales modificadas mediante una transformación lineal, A (Figura 2.1). El problema consiste en obtener los coeficientes de la aplicación lineal, a_{ij} , y a partir de ellos calcular la inversa de la transformación. Esto hace posible la reconstrucción de las señales originales, s_j , conociendo sólo las señales observadas, e_i .

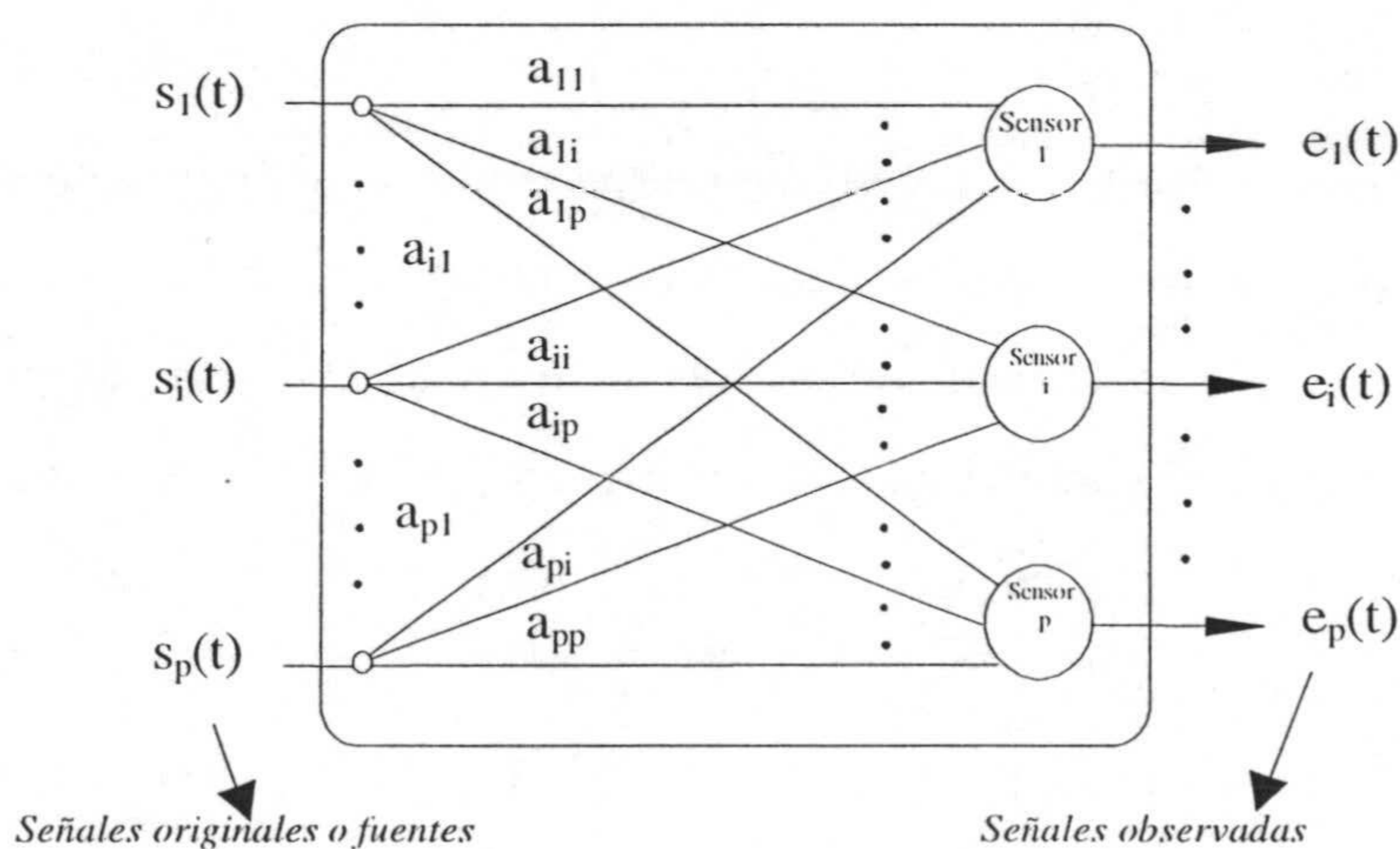


Figura 2.1. Separación ciega de fuentes.

Como se vio en el Capítulo 1, en la literatura se pueden encontrar distintas aproximaciones y la mayoría de ellas se basan en criterios estadísticos o de la Teoría Estadística de la

Información (Sec.1.3), y algunos de estos algoritmos están implementados con redes neuronales artificiales. En esta memoria se proponen métodos geométricos, aplicables a cualquier tipo de señales que cumplan las condiciones indicadas en la Sección 2.3, generadas por dos o más fuentes ($p \geq 2$), que producen la separación analizando las fuentes observadas en un espacio p -dimensional. Estos procedimientos se pueden implementar utilizando técnicas clásicas de teoría de señales, o con redes neuronales artificiales, consiguiendo de esta última forma las ventajas inherentes a este tipo de redes, tales como regularidad, paralelismo y funcionamiento en tiempo real [PRI96, HAY99].

2.1.1 Hipótesis del procedimiento.

Las principales hipótesis que se utilizan en los métodos geométricos son las siguientes:

Hipótesis 2.1. Las señales $s_j(t)$, ($j = 1, \dots, p$), y sus transformaciones lineales, $e_j(t)$, son desconocidas.

Hipótesis 2.2. El número de sensores es igual al número de fuentes, p .

Hipótesis 2.3. Las fuentes están acotadas:

$$s_{im} \leq s_i(t) \leq s_{iM} \quad \begin{array}{l} s_{im} \leq s_{iM} \\ s_{im}, s_{iM} \in \mathfrak{R} \end{array} \quad (2.1)$$

Podemos hacer esta restricción ya que en realidad, las señales físicas (voz, radar, sonar, señales biomédicas, etc.) están limitadas en amplitud. Además, esta hipótesis es razonable ya que cualquier señal producida por una fuente real y transmitida a través de un medio físico debe tener una energía finita.

Hipótesis 2.4. Las señales observadas, $e=(e_i(t))^T$, se obtienen a partir de las fuentes, $s=(s_i(t))^T$, mediante la transformación lineal, $A=(a_{ij})$, y por tanto:

$$e(t) = A \cdot s(t) \quad \text{con:} \quad \begin{array}{l} e, s \in \mathfrak{R}^p \\ A \in \mathfrak{R}^{p \times p} \end{array} \quad (2.2)$$

donde $e(t)$ y $s(t)$ son vectores de componentes $e_i(t)$ y $s_i(t)$ respectivamente, que en (2.2) aparecen como matrices columna. La transformación lineal A es una matriz $p \times p$ desconocida (*matriz de mezcla*).

Hipótesis 2.5. En algunos de los métodos geométricos se supone que todos los elementos de la diagonal principal de la matriz A son distintos de cero:

$$a_{ii} \neq 0, \quad \forall i \in \{1, \dots, p\} \quad (2.3)$$

Esta condición es razonable, ya que indica que cada sensor i , asociado a la fuente de su mismo índice i , detecta la señal procedente de esta ($s_i(t)$).

Si se supone la no existencia de retardos entre las señales detectadas y las fuentes, se dice que la ecuación (2.2) modela una mezcla lineal instantánea (Sec.1.1).

El problema de separación consiste en recuperar las fuentes desconocidas $s(t)$, sólo a partir de las observaciones $e(t)$. Si, por algún procedimiento, se puede obtener A , resulta inmediato recuperar las señales originales, a partir de la siguiente expresión:

$$s(t) = A^{-1} \cdot e(t) \quad (2.4)$$

Sin embargo, es imposible determinar exactamente A conociendo sólo las señales mezcladas. No obstante, en el campo de la separación ciega de fuentes (Sec.1.2) se considera una solución válida la recuperación de las señales en la forma $c_i \cdot s_i$, $i \in \{1, \dots, p\}$. Esta recuperación consiste en obtener las señales originales multiplicadas por una constante c_i que representa un factor de escala indefinido y distinto de cero. Físicamente, este factor corresponde a una amplificación ($c_i > 1$) o atenuación ($c_i < 1$) de las señales, con inversión ($c_i < 0$) o sin ella ($c_i > 0$). También se considera una solución válida si los índices de las señales separadas están permutados respecto a los de las fuentes originales. Las señales recuperadas, identificables con las indeterminaciones indicadas las denominamos

fuentes estimadas o recuperadas: $y=(c_i \cdot s_i)^T$. Analíticamente se pueden representar las dos indeterminaciones de la siguiente forma:

$$y(t) = \mathbf{D} \cdot \mathbf{Z} \cdot s(t) \quad (2.5)$$

donde \mathbf{Z} es una matriz de permutación de índices:

$$\begin{aligned} \mathbf{Z} &= (z_{ij}) \\ z_{ij} &= \{0,1\} \\ \sum_{i=1}^p z_{ij} &= 1 \quad \sum_{j=1}^p z_{ij} = 1 \\ i, j &\in \{1,2,\dots, p\} \end{aligned} \quad (2.6)$$

y \mathbf{D} una matriz diagonal de factores de escala:

$$\begin{aligned} \mathbf{D} &= (d_{ij}) \\ d_{ij} &= c_i \cdot \delta_{ij} \\ \delta_{ii} &= 1 \quad \delta_{ij} = 0 \quad \forall i \neq j \\ i, j &\in \{1,\dots, p\} \end{aligned} \quad (2.7)$$

Matemáticamente las señales $y(t)$ se pueden obtener a partir de las señales mezcladas, $e(t)$, en términos de una transformación lineal \mathbf{W} :

$$y(t) = \mathbf{W}^{-1} \cdot e(t) \quad (2.8)$$

sustituyendo (2.2) en (2.8):

$$y(t) = \mathbf{W}^{-1} \cdot \mathbf{A} \cdot s(t) \quad (2.9)$$

identificando (2.5) y (2.9):

$$\mathbf{W}^{-1} \cdot \mathbf{A} = \mathbf{D} \cdot \mathbf{Z} \stackrel{def}{=} \mathbf{P} \quad (2.10)$$

Definición 2.1. Cualquier matriz W relacionada con A según (2.10) se dice que es *similar a* A , o constituye una estimación de A ($W = \tilde{A}$).

Cualquier matriz similar a A se considera válida para obtener las fuentes transformadas, $y(t)$. Si las fuentes originales se obtienen sin un factor de escala o índice de permutación la matriz P definida en (2.10) sería $P = I$ donde $I = (\delta_{ij})$. Denominamos a la matriz P *matriz de prestaciones* por motivos que más adelante se verán.

En general, cuando se tiene en cuenta el factor de escala, P no es una matriz unidad. Por ejemplo, para $p=3$:

$$\mathbf{D} = \begin{bmatrix} c_1 & 0 & 0 \\ 0 & c_2 & 0 \\ 0 & 0 & c_3 \end{bmatrix} \quad \text{y} \quad \mathbf{Z} = \begin{bmatrix} z_{11} & z_{12} & z_{13} \\ z_{21} & z_{22} & z_{23} \\ z_{31} & z_{32} & z_{33} \end{bmatrix}$$

$$\text{con: } z_{ij} = \{0,1\} \quad \text{y} \quad \sum_{i=1}^p z_{ij} = 1 \quad \text{y} \quad \sum_{j=1}^p z_{ij} = 1$$

$$\text{si tomamos: } \quad \mathbf{Z} = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \quad \text{entonces:}$$

$$\mathbf{P} = \begin{bmatrix} 0 & 0 & c_1 \\ c_2 & 0 & 0 \\ 0 & c_3 & 0 \end{bmatrix} \quad (2.11)$$

Si se obtiene la matriz W iterativamente, no se consigue la separación completa desde las primeras iteraciones. En esta situación y suponiendo el caso anterior (2.11), inicialmente la matriz P podría ser de la siguiente forma:

$$\mathbf{P} = \begin{bmatrix} \varepsilon_{12} & \varepsilon_{13} & \varepsilon_{11} \cdot c_1 \\ \varepsilon_{22} \cdot c_2 & \varepsilon_{23} & \varepsilon_{21} \\ \varepsilon_{32} & \varepsilon_{33} \cdot c_3 & \varepsilon_{31} \end{bmatrix} \quad (2.12)$$

Los valores de ε_{ij} , $i \neq j$, irán tendiendo a cero y los de ε_{ij} , $i=j$, a uno a medida que el proceso iterativo va convergiendo. En consecuencia, en el proceso iterativo inicialmente no se obtiene una matriz W similar a A , teniendo P la forma (2.12). Conforme avanza el proceso iterativo, si hay convergencia $\varepsilon_{ij} \rightarrow 0$, $\varepsilon_{ii} \rightarrow 1$, y P va tomando la forma (2.11), es decir, se llega a verificar (2.10) haciéndose W similar a A .

Los métodos geométricos de separación se basan en las propiedades geométricas y algebraicas de la transformación que tiene lugar entre las fuentes y las mezclas. Estos algoritmos se caracterizan por su sencillez, requiriendo (en algunos de ellos) que las fuentes produzcan, al menos una vez, combinaciones de sus valores extremos. En estas condiciones los procedimientos geométricos propuestos en esta memoria pueden realizar la separación incluso de señales estadísticamente dependientes.

La hipótesis de que las fuentes sean acotadas implica que los posibles vectores fuente, $s(t)$, ocupen una zona del espacio geoméricamente acotada, el *espacio fuente* S , dentro del espacio (s_1, \dots, s_p) . Si las fuentes fueran estadísticamente independientes, por ejemplo para $p=2$, el espacio fuente sería un rectángulo, o en general para cualquier dimensión, un hiperparalelepípedo. La transformación lineal A proyecta los vectores del espacio fuente en vectores (e_1, \dots, e_p) , que forman el *espacio de mezclas* E . Se puede probar (Sec.2.2) que si el espacio fuente es un hiperparalelepípedo entonces, el espacio de mezclas también lo es.

2.1.2 Modelo con ruido aditivo

El ruido puede considerarse, dentro del problema de separación de fuentes bajo tres perspectivas distintas:

1. *El ruido incide en el medio como una señal más.* En este caso los algoritmos que se proponen consideran al ruido como una de las p señales y lo representan correctamente como si se tratara de una fuente cualquiera (ver Sec.4.7).
2. *Las señales originales están afectadas por ruido.* En esta situación los algoritmos de separación actúan transparentemente con respecto al ruido, sin incrementarlo ni reducirlo, recuperándose las señales con la misma relación señal-ruido original.

3. *El ruido corrompe las señales ya mezcladas, en el interior del medio o de los sensores.*
Este es el único caso que debe analizarse ya que para reducirlo los algoritmos deben tenerlo en cuenta. Tradicionalmente, para modelar este caso, únicamente se considera el caso más sencillo de ruido aditivo y acotado.

En este último caso, el modelo matemático vectorial tomado para las mezclas sería el siguiente:

$$e_i^{(c)}(t) = e_i(t) + n_i(t) = \sum_{j=1}^p a_{ij} \cdot s_j(t) + n_i(t) \quad j = 1, \dots, p \quad (2.13)$$

donde denotamos con $e_i^{(c)}$ los vectores mezcla corrompidos o afectados de ruido aditivo $n_i(t) \subset [n_{im}, n_{iM}]$, siendo n_{im} y n_{iM} las cotas de error con $n_{im} < n_{iM}$. Es usual modelar el ruido, $\mathbf{n}(t)$, con [MIT93]:

- a) una señal aleatoria uniformemente distribuida,
- b) una señal Gaussiana, o
- c) una señal Laplaciana.

O matricialmente:

$$\mathbf{e}^{(c)}(t) = \mathbf{A} \cdot \mathbf{s}(t) + \mathbf{n}(t) = \mathbf{e}(t) + \mathbf{n}(t) \quad \text{con} \quad \mathbf{n}(t) \stackrel{\text{def}}{=} (n_i(t))^T \quad (2.14)$$

Donde la relación señal-ruido correspondiente a la mezcla $e_i(t)$ sería:

$$SNR(e_i) = 10 \cdot \log \left(\frac{e_i^{(c)}(t)}{n_i(t)} \right)^2 = 20 \cdot \log \left(\frac{e_i^{(c)}(t)}{n_i(t)} \right) \quad (\text{dBs}) \quad (2.15)$$

El ruido afectará a la recuperación de las señales de dos formas:

1. No se obtendrá una matriz \mathbf{W} correcta, sino otra $\mathbf{W}^{(c)}$ corrompida, ya que el algoritmo de obtención de \mathbf{W} utiliza los vectores $\mathbf{y}^{(c)}(t)$ en lugar de los $\mathbf{y}(t)$.
2. Las señales recuperadas serán de la forma:

$$\begin{aligned}
 y^{(c)}(t) &= [W^{(c)}]^{-1} \cdot e^{(c)}(t) = [W^{(c)}]^{-1} \cdot [e(t) + n(t)] = \\
 &= [W^{(c)}]^{-1} \cdot e(t) + [W^{(c)}]^{-1} \cdot n(t)
 \end{aligned}
 \tag{2.16}$$

En las Secciones 4.4 y 4.7 se incluyen procedimientos para reducir la influencia del ruido sobre las mezclas en el proceso de recuperación de las señales originales.

2.2 PROPIEDADES ANALÍTICAS Y GEOMÉTRICAS DE LA TRANSFORMACIÓN LINEAL DE LAS FUENTES.

En esta sección se incluyen una serie de propiedades relacionadas con la transformación A . Estas propiedades sirven de fundamento a la Sec.2.3 donde se describirán los principios del método geométrico.

Para simplificar la comprensión de estas propiedades a veces sólo se considera el caso de dos fuentes ($p=2$), siendo inmediatamente generalizables, para $p>2$, tanto los enunciados de las propiedades como sus demostraciones.

En el caso de tener únicamente dos fuentes, es decir, $p=2$, la mezcla de señales (2.2) viene dada por:

$$\begin{aligned}
 e_1(t) &= a_{11} \cdot s_1(t) + a_{12} \cdot s_2(t) \\
 e_2(t) &= a_{21} \cdot s_1(t) + a_{22} \cdot s_2(t)
 \end{aligned}
 \tag{2.17}$$

Al estar las fuentes acotadas, es decir, $s_{im} \leq s_i \leq s_{iM}$, los valores generados por las mismas estarán circunscritos, tal y como se muestra en la Figura 2.2a, en el interior de un rectángulo.

Obviamente, los vértices de este rectángulo son:

$$\begin{aligned}
 \text{vértice } v_{s0} &\rightarrow (s_{1m}, s_{2m}) \\
 \text{vértice } v_{s1} &\rightarrow (s_{1M}, s_{2m}) \\
 \text{vértice } v_{s2} &\rightarrow (s_{1m}, s_{2M}) \\
 \text{vértice } v_{s3} &\rightarrow (s_{1M}, s_{2M})
 \end{aligned}
 \tag{2.18}$$

Por otra parte las aristas son segmentos incluidos en las rectas definidas por las siguientes ecuaciones:

$$\begin{aligned}
 \text{recta } [v_{s0}, v_{s1}] & \quad s_2 = s_{2m} \\
 \text{recta } [v_{s0}, v_{s2}] & \quad s_1 = s_{1m} \\
 \text{recta } [v_{s2}, v_{s3}] & \quad s_2 = s_{2M} \\
 \text{recta } [v_{s1}, v_{s3}] & \quad s_1 = s_{1M}
 \end{aligned} \tag{2.19}$$

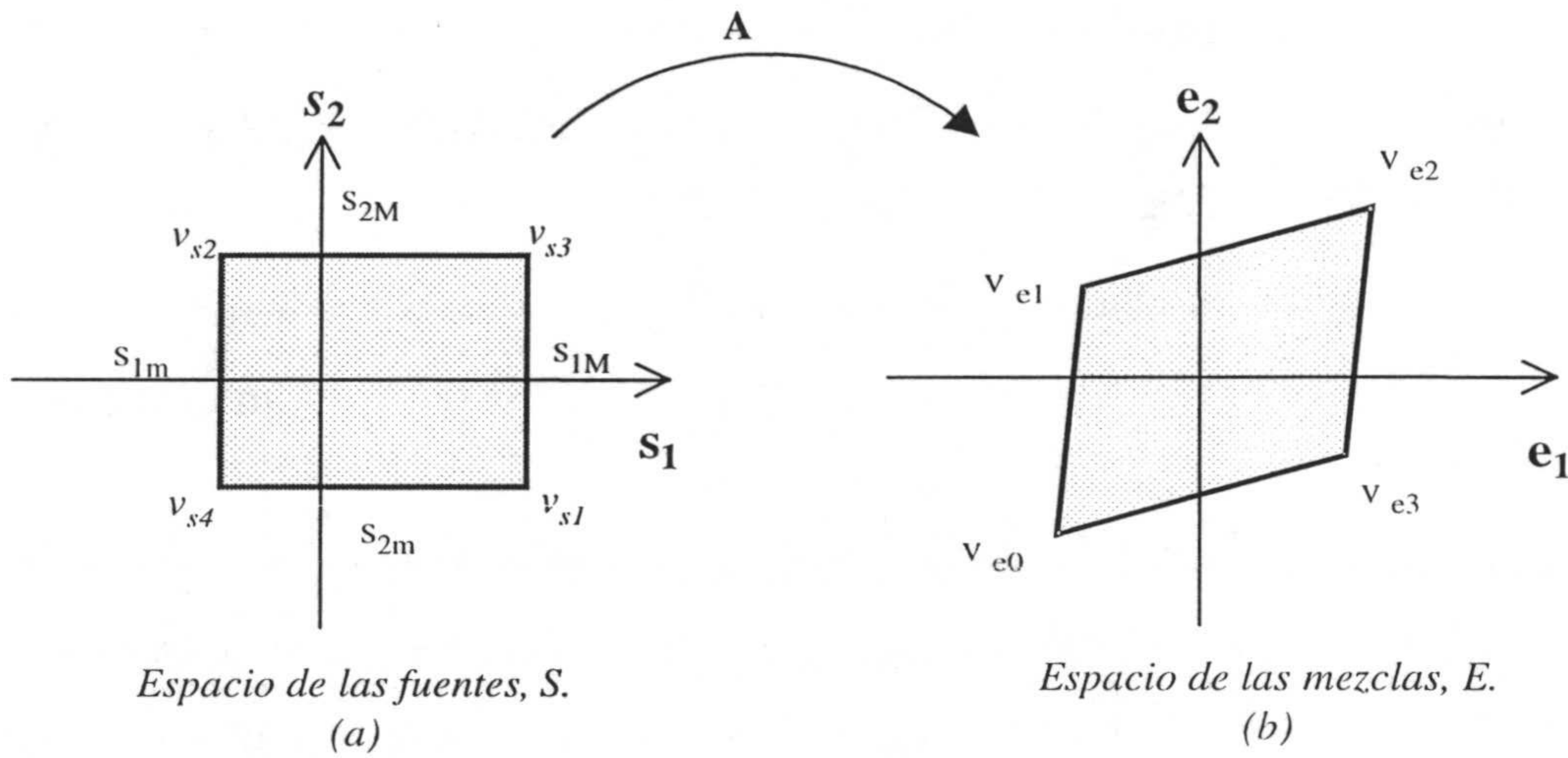


Figura 2.2. Espacio fuente y espacio de las mezclas para $p=2$.

Las imágenes de los vértices $(v_{s0}, v_{s1}, v_{s2}, v_{s3})$, en el espacio de observaciones E, $(e^{[vs0]}, e^{[vs1]}, e^{[vs2]}, e^{[vs3]})$ pueden obtenerse a partir de las ecuaciones (2.20) (Figura 2.2b):

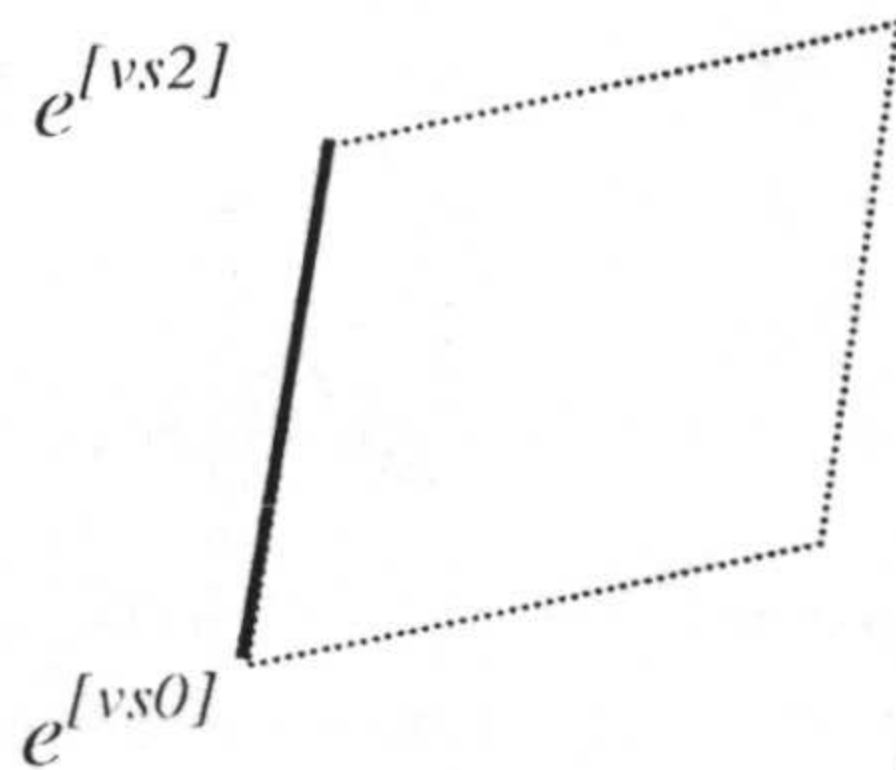
$$\begin{aligned}
 \text{vértice } e^{[vs0]} & \rightarrow (s_{1m}, s_{2m}) : & \text{vértice } e^{[vs2]} & \rightarrow (s_{1m}, s_{2M}) : \\
 e_1^{[vs0]} & = a_{11} \cdot s_{1m} + a_{12} \cdot s_{2m} & e_1^{[vs2]} & = a_{11} \cdot s_{1m} + a_{12} \cdot s_{2M} \\
 e_2^{[vs0]} & = a_{21} \cdot s_{1m} + a_{22} \cdot s_{2m} & e_2^{[vs2]} & = a_{21} \cdot s_{1m} + a_{22} \cdot s_{2M} \\
 \text{vértice } e^{[vs1]} & \rightarrow (s_{1M}, s_{2m}) : & \text{vértice } e^{[s2]} & \rightarrow (s_{1M}, s_{2M}) : \\
 e_1^{[vs1]} & = a_{11} \cdot s_{1M} + a_{12} \cdot s_{2m} & e_1^{[vs3]} & = a_{11} \cdot s_{1M} + a_{12} \cdot s_{2M} \\
 e_2^{[vs1]} & = a_{21} \cdot s_{1M} + a_{22} \cdot s_{2m} & e_2^{[vs3]} & = a_{21} \cdot s_{1M} + a_{22} \cdot s_{2M}
 \end{aligned} \tag{2.20}$$

Donde $e_i^{[vsj]}$ representa el punto imagen en el espacio de observaciones del vértice v_{sj} del espacio de las fuentes.

Los vértices definen un paralelepípedo cuyas aristas vienen determinadas por las ecuaciones (2.22) a (2.25), donde:

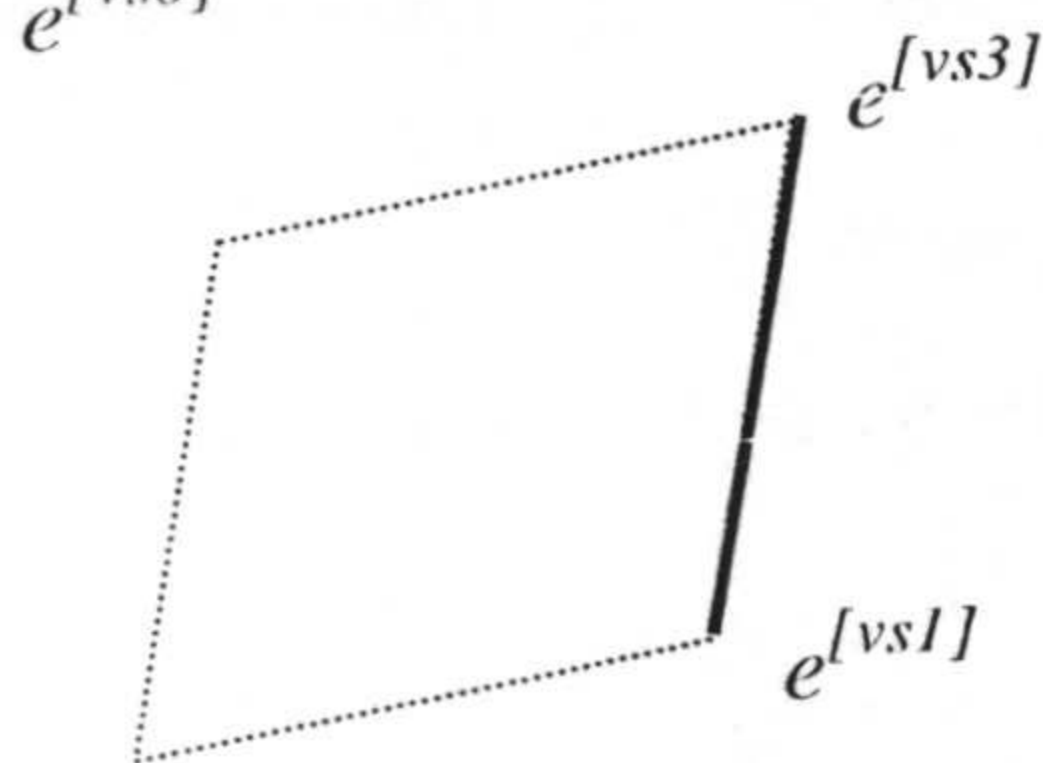
$$\Delta_A = \det(A) = a_{11} \cdot a_{22} - a_{12} \cdot a_{21} \quad (2.21)$$

Las ecuaciones de las rectas que pasan por las aristas son:



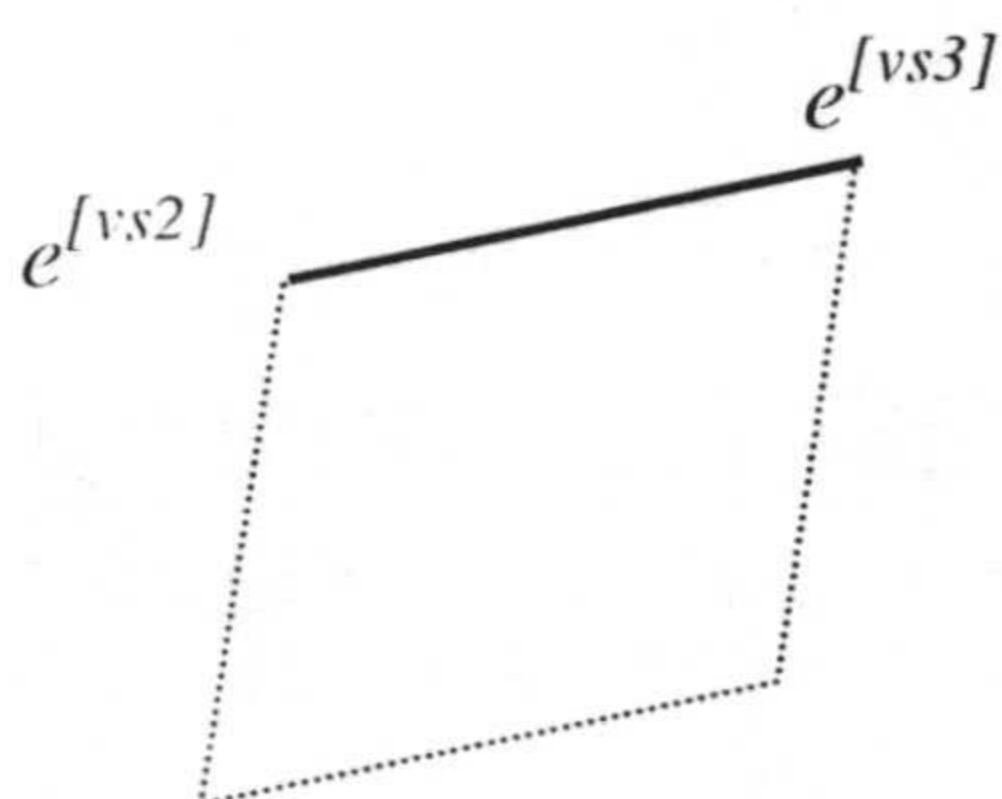
Recta $[e^{[vs0]}, e^{[vs2]}]$:

$$e_2 = \frac{a_{22}}{a_{12}} \cdot e_1 - \frac{\Delta_A}{a_{12}} \cdot s_{1m} \quad (2.22)$$



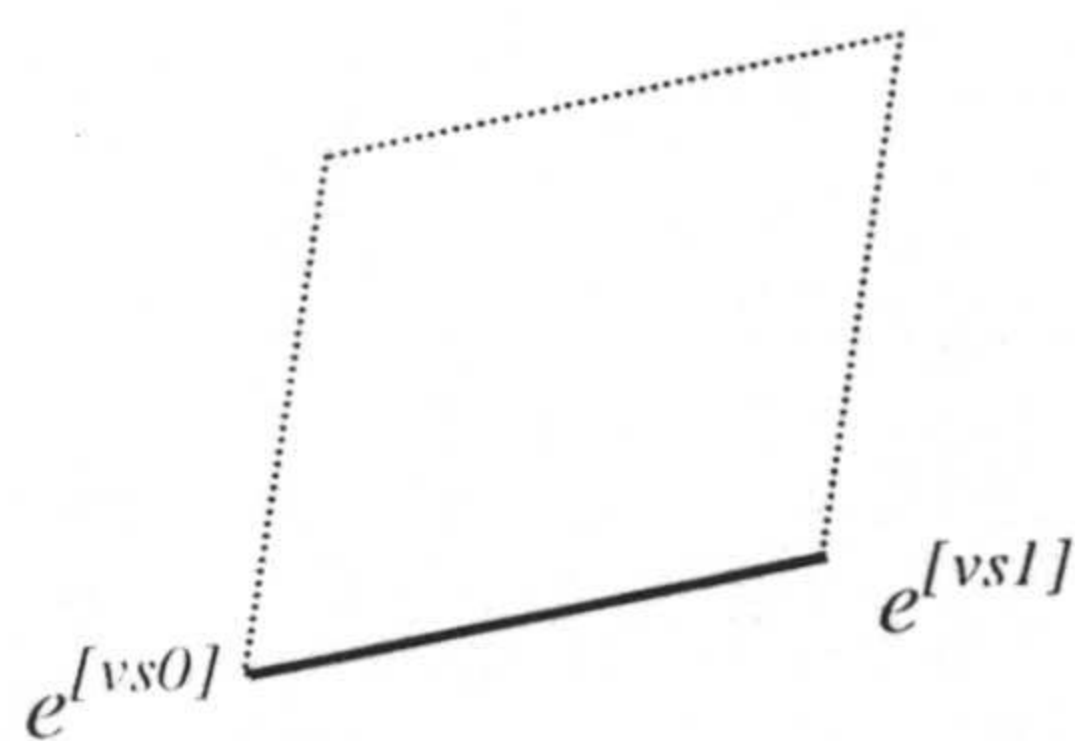
Recta $[e^{[vs1]}, e^{[vs3]}]$:

$$e_2 = \frac{a_{22}}{a_{12}} \cdot e_1 - \frac{\Delta_A}{a_{12}} \cdot s_{1M} \quad (2.23)$$



Recta $[e^{[vs2]}, e^{[vs3]}]$:

$$e_2 = \frac{a_{21}}{a_{11}} \cdot e_1 + \frac{\Delta_A}{a_{11}} \cdot s_{2m} \quad (2.24)$$



Recta $[e^{[vs0]}, e^{[vs1]}]$:

$$e_2 = \frac{a_{21}}{a_{11}} \cdot e_1 + \frac{\Delta_A}{a_{11}} \cdot s_{2M} \quad (2.25)$$

A continuación se generalizan los resultados anteriores para el caso de más de dos fuentes ($p > 2$). Para ello se denomina $s_i^{[s_j]}$ a la coordenada i del vértice v_{sj} . Los valores de estas componentes de los 8 vértices para $p=3$ en función de las cotas $[s_{jm}, s_{jM}]$ de las distintas componentes pueden verse en la Tabla 2.1.

TABLA 2.1. Condición de vértice para un espacio de fuentes cúbico ($p=3$)

j	$s_3^{[s_j]}$	$s_2^{[s_j]}$	$s_1^{[s_j]}$	Condición de vértice
0	s_{3m}	s_{2m}	s_{1m}	máx(- $s_3-s_2-s_1$)
1	s_{3m}	s_{2m}	s_{1M}	máx(- $s_3-s_2+s_1$)
2	s_{3m}	s_{2M}	s_{1m}	máx(- $s_3+s_2-s_1$)
3	s_{3m}	s_{2M}	s_{1M}	máx(- $s_3+s_2+s_1$)
4	s_{3M}	s_{2m}	s_{1m}	máx(+ $s_3-s_2-s_1$)
5	s_{3M}	s_{2m}	s_{1M}	máx(+ $s_3-s_2+s_1$)
6	s_{3M}	s_{2M}	s_{1m}	máx(+ $s_3+s_2-s_1$)
7	s_{3M}	s_{2M}	s_{1M}	máx(+ $s_3+s_2+s_1$)

También se incluye la condición que deben cumplir las coordenadas de un punto del espacio de las fuentes para ser vértice en la tercera columna.

Los puntos imagen de los vértices, v_{sj} , del espacio S en el espacio E, $e^{[vsj]}$, se pueden obtener sin más que sustituir las componentes de v_{sj} en la expresión (2.2), obteniéndose así:

$$e^{[vsj]} = (e_i^{[vsj]}) \quad \text{con} \quad e_i^{[vsj]} = \sum_{k=1}^p a_{ik} \cdot s_k^{[sj]} \quad (2.26)$$

Estos puntos determinan un espacio cerrado que constituye un hiperparalelepípedo. Dos vértices adyacentes, es decir, aquéllos cuyos vértices origen difieren en una componente, definen una arista del hiperparalelepípedo.

Si, por ejemplo, se considera el vértice:

$$e^{[vs_0]} = (e_i^{[vs_0]}) \quad \text{con} \quad e_i^{[vs_0]} = \sum_{k=1}^p a_{ik} \cdot s_{km} \quad (2.27)$$

los vértices adyacentes a $e^{[vs_0]}$ serán: $e^{[vs_1]}$ (sus orígenes se diferencian sólo en la componente s_1), $e^{[vs_2]}$ (sus orígenes se diferencian sólo en la componente s_2) y $e^{[vs_4]}$ (sus orígenes se diferencian sólo en la componente s_3), etc. (ver Tabla 2.1).

Denominamos: *arista 1*, a la arista formada por los vértices $e^{[vs0]}$ y $e^{[vs1]}$; *arista 2*, a la arista formada por los vértices $e^{[vs0]}$ y $e^{[vs2]}$; *arista 3*, a la arista formada por los vértices $e^{[vs0]}$ y $e^{[vs4]}$; y en general *arista j*, a la arista formada por los vértices $e^{[vs0]}$ y $e^{[vsn]}$, siendo $n=2^{(j-1)}$.

Pueden obtenerse con facilidad las ecuaciones de las aristas, expresándolas en forma de "ecuación de una recta que pasa por dos puntos". Así, la ecuación de la arista 3 (es decir, la recta que pasa por los puntos $e^{[vs0]}$ y $e^{[vs4]}$) será:

$$\frac{e_1 - e_1^{[vs0]}}{e_1^{[vs4]} - e_1^{[vs0]}} = \frac{e_2 - e_2^{[vs0]}}{e_2^{[vs4]} - e_2^{[vs0]}} = \dots = \frac{e_p - e_p^{[vs0]}}{e_p^{[vs4]} - e_p^{[vs0]}} \quad (2.28)$$

Teniendo en cuenta (2.26) y la Tabla 2.1, puede simplificarse el denominador:

$$\frac{e_1 - e_1^{[vs0]}}{a_{13} \cdot s_{3M} - a_{13} \cdot s_{3m}} = \frac{e_2 - e_2^{[vs0]}}{a_{23} \cdot s_{3M} - a_{23} \cdot s_{3m}} = \dots = \frac{e_p - e_p^{[vs0]}}{a_{p3} \cdot s_{3M} - a_{p3} \cdot s_{3m}} \quad (2.29)$$

y multiplicando todos los miembros por $(s_{3M} - s_{3m})$, resulta:

$$\frac{e_1 - e_1^{[vs0]}}{a_{13}} = \frac{e_2 - e_2^{[vs0]}}{a_{23}} = \dots = \frac{e_p - e_p^{[vs0]}}{a_{p3}} \quad (2.30)$$

Es decir, la arista 3 tiene las siguientes pendientes:

$$\begin{aligned} \text{plano } 2,1: \quad & \frac{\Delta e_2}{\Delta e_1} = \frac{a_{23}}{a_{13}} \\ \text{plano } 3,1: \quad & \frac{\Delta e_3}{\Delta e_1} = \frac{a_{33}}{a_{13}} \\ \dots & \dots \\ \text{plano } p,1: \quad & \frac{\Delta e_p}{\Delta e_1} = \frac{a_{p3}}{a_{13}} \end{aligned} \quad (2.31)$$

Se puede concluir, por tanto, de las expresiones (2.22) a (2.25) (correspondientes a $p=2$) y de (2.31) (para $p \geq 2$) la siguiente propiedad.

Propiedad 2.1: La arista j del hiperparalelepípedo en el espacio de observaciones, incidente en el vértice $e^{[vs0]}$, tiene de pendiente en el plano k, i :

$$\frac{\Delta e_k}{\Delta e_i} = \frac{a_{kj}}{a_{ij}} \quad \forall k, i, j \in \{0, 1, \dots, p\} \quad (2.32)$$

Obsérvese que según (2.30), si el hiperparalelepípedo del espacio de observaciones se traslada al vértice $e^{[vs0]}$ las ecuaciones de las aristas incidentes en dicho vértice serían:

$$\begin{aligned} \text{arista } 1: & \quad \frac{e_1}{a_{11}} = \frac{e_2}{a_{21}} = \dots = \frac{e_p}{a_{p1}} \\ \text{arista } 2: & \quad \frac{e_1}{a_{12}} = \frac{e_2}{a_{22}} = \dots = \frac{e_p}{a_{p2}} \\ & \dots\dots\dots \\ \text{arista } p: & \quad \frac{e_1}{a_{1p}} = \frac{e_2}{a_{2p}} = \dots = \frac{e_p}{a_{pp}} \end{aligned} \quad (2.33)$$

Propiedad 2.2: El espacio interior al paralelepípedo $[e^{[vs0]}, e^{[vs1]}, e^{[vs2]}, e^{[vs3]}]$ constituye el espacio de observaciones E ; es decir, todo punto del espacio fuente S (interior al paralelepípedo $[v_{s0}; v_{s1}; v_{s2}; v_{s3}]$) tiene como imagen un punto interior al paralelepípedo $[e^{[vs0]}, e^{[vs1]}, e^{[vs2]}, e^{[vs3]}]$.

Formalmente la propiedad anterior (para $p=2$) se puede expresar como:

$$si (s_1, s_2) \mid s_{1m} \leq s_1 \leq s_{1M} \text{ y } s_{2m} \leq s_2 \leq s_{2M} \Rightarrow (e_1, e_2) \mid e_1^{[0;2]} \leq e_1 \leq e_1^{[1;3]} \text{ y } e_2^{[0;1]} \leq e_2 \leq e_2^{[2;3]}$$

siendo (Figura 2.3):

- $(e_1^{[0;2]}, e_2)$ las coordenadas de un punto genérico de la arista $[e^{[vs0]}, e^{[vs2]}]$,
- $(e_1^{[1;3]}, e_2)$ las coordenadas de un punto genérico de la arista $[e^{[vs1]}, e^{[vs3]}]$,
- $(e_1, e_2^{[0;1]})$ las coordenadas de un punto genérico de la arista $[e^{[vs0]}, e^{[vs1]}]$ y

$(e_1, e_2^{[2;3]})$ las coordenadas de un punto genérico de la arista $[e^{[vs2]}, e^{[vs3]}]$

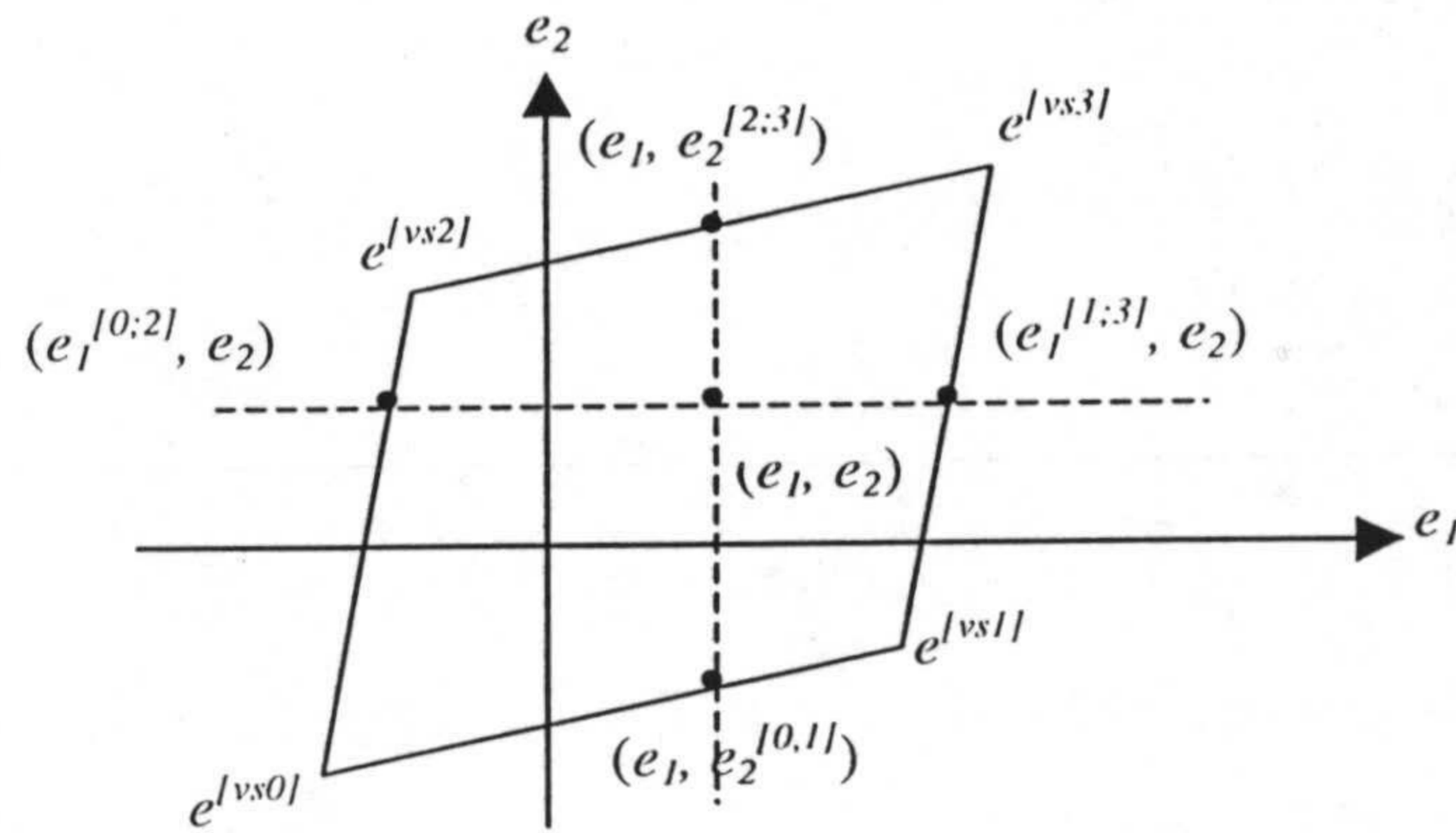


Figura 2.3. Paralelepípedo del espacio de observaciones.

Así, se puede probar, por ejemplo, que:

$$s_1 \leq s_{1M} \Rightarrow e_1 \leq e_1^{[1;3]}$$

En efecto, de (2.22) se tiene que:

$$e_1^{[1;3]} = \frac{a_{12}}{a_{22}} \cdot e_2 + \frac{\Delta_A}{a_{22}} \cdot s_{1M} \quad (2.34)$$

Por otro lado de (2.17):

$$s_1 = \frac{a_{22}}{\Delta_A} \cdot e_1 - \frac{a_{12}}{\Delta_A} \cdot e_2 \quad (2.35)$$

y si $s_1 \leq s_{1M}$ se verificará que:

$$\frac{a_{22}}{\Delta_A} \cdot e_1 - \frac{a_{12}}{\Delta_A} \cdot e_2 \leq s_{1M} \quad (2.36)$$

o de forma similar:

$$e_1 \leq \frac{\Delta_A}{a_{22}} \cdot s_{1M} + \frac{a_{12}}{a_{22}} \cdot e_2 \quad (2.37)$$

con lo que, teniendo en cuenta (2.34):

$$e_1 \leq e_1^{[1;3]}$$

que es lo que se deseaba probar.

De la misma forma se pueden demostrar las otras 3 desigualdades que implica la Propiedad 2.1, siendo válido el enunciado de la misma.

Propiedad 2.3: *La imagen de un vértice del espacio de fuentes S es un vértice del espacio de observaciones E .*

Esta propiedad resulta evidente por la forma de construir el paralelepípedo $[v_{e0}; v_{e1}; v_{e2}; v_{e3}]$ y la Propiedad 2.1.

Propiedad 2.4: *La matriz, W , resultante de multiplicar una columna j de la matriz A por una constante c_j es similar a A , y la fuente y_j reconstruida está relacionada con la fuente original s_j según la relación:*

$$y_j = \frac{1}{c_j} \cdot s_j \quad (2.38)$$

Para simplificar la demostración de esta propiedad supóngase que $p=2$ y que se multiplica por la constante c_j la primera columna de A . Se tiene, que:

$$W = \begin{pmatrix} c_1 \cdot a_{11} & a_{12} \\ c_1 \cdot a_{21} & a_{22} \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} \cdot \begin{pmatrix} c_1 & 0 \\ 0 & 1 \end{pmatrix} = A \cdot D^* \quad (2.39)$$

con lo que (2.39) también se puede escribir:

$$W^{-1} \cdot A = D \cdot Z \quad (2.40)$$

siendo:

$$D = (D^*)^{-1} = \begin{pmatrix} \frac{1}{c_1} & 0 \\ 0 & 1 \end{pmatrix} \quad Z = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \quad (2.41)$$

y, por tanto, según la Definición 2.1, W es equivalente a A .

De (2.40) y teniendo en cuenta que $Z = I$ se obtiene que:

$$W^{-1} \cdot A = D \quad (2.42)$$

Dado el valor de D en (2.41) y que $y(t) = W^{-1} A \cdot s(t)$, resulta:

$$\begin{aligned} y_1(t) &= \frac{1}{c_1} \cdot s_1(t) \\ y_2(t) &= s_2(t) \end{aligned} \quad (2.43)$$

como se quería probar.

Con ayuda de estas sencillas propiedades se pueden comprender fácilmente los principios de los métodos geométricos de separación de fuentes, tal y como han sido enunciados por Puntonet y Prieto [PUN95, PUN98, PRI98, PRI99].

2.3 FUNDAMENTO DEL MÉTODO GEOMÉTRICO DE SEPARACIÓN DE FUENTES.

Según se ha indicado anteriormente el problema de separación de fuentes, con las hipótesis indicadas en la Sec.2.1.1 consiste en encontrar una matriz W similar a la matriz A , partiendo sólo de vectores de observación $e(t)$. En esta sección se demostrarán tres propiedades que constituyen la base tanto de los distintos algoritmos geométricos propuestos hasta ahora, como de los que se plantean en los siguientes capítulos.

Propiedad 2.5: La matriz $W=(w_{ij})$ obtenida a partir de las pendientes de las p aristas incidentes en el vértice $e^{[vs0]}$, tal que:

$$w_{ij} = \text{pendiente en el plano } (i, j) \text{ de la arista } j$$

es similar a A .

En efecto, según el enunciado de la propiedad:

$$w_{ij} = \frac{\Delta e_i}{\Delta e_j} \quad (\text{arista } j) \quad (2.44)$$

y según la Propiedad 2.1:

$$w_{ij} = \frac{\Delta e_i}{\Delta e_j} = \frac{a_{ij}}{a_{jj}} \quad (\text{arista } j) \quad (2.45)$$

Por consiguiente, se puede relacionar W con A según la expresión:

$$A = W \cdot D \cdot Z \quad (2.46)$$

donde $P=DZ=I$ y $c_i = a_{ii}$, y por tanto:

$$y_i = \frac{1}{a_{ii}} \cdot s_i \quad i = \{1, 2, \dots, p\} \quad (2.47)$$

Si $a_{ii} < 0$ la señal reconstruida estará invertida respecto a la original, si $a_{ii} > 1$ estará atenuada, y si $a_{ii} < 1$ estará amplificada. Recuérdese que se está haciendo uso de una de las hipótesis (Hipótesis 2.5, Sec.2.1.1), $a_{ii} \neq 0$, ya que si no se verificase la misma no se podría reconstruir la señal s_i , por resultar $y_i = \infty$.

Definición 2.2:

Denominamos **vectores-aristas incidentes** en un vértice v_{ex} , a un conjunto cualquiera de p vectores (o puntos) cada uno de ellos perteneciente a una de las p diferentes aristas del cono del hiperparalelepípedo con origen en v_{ex} que contiene al espacio de observaciones .

Propiedad 2.6:

La matriz, $W=(w_{ij})$, obtenida con un conjunto de p vectores-aristas, $w^{[j]}$, incidentes en el vértice $e^{[vs0]}$, considerado como origen, es similar a A . La matriz W se forma tomando cada vector-arista como columna de W , es decir:

$$W = [w^{[1]} \dots w^{[j]} \dots w^{[p]}] \tag{2.48}$$

En efecto, obsérvese según la Propiedad 2.1, que al considerar $e^{[vs0]}$ como origen, la ecuación de la arista j puede expresarse como:

$$\frac{e_1}{a_{1j}} = \frac{e_2}{a_{2j}} = \dots = \frac{e_p}{a_{pj}} \tag{2.49}$$

y sus ecuaciones paramétricas resultan ser:

$$\begin{aligned} e_1 &= t_j & t_j &\in \mathfrak{R} \\ e_2 &= \frac{a_{2j}}{a_{1j}} \cdot t_j \\ &\dots\dots\dots \\ e_i &= \frac{a_{ij}}{a_{1j}} \cdot t_j \\ &\dots\dots\dots \\ e_p &= \frac{a_{pj}}{a_{1j}} \cdot t_j \end{aligned} \tag{2.50}$$

Con lo que la coordenada j de un punto genérico de la arista- j es:

$\left(t_j, \frac{a_{2j}}{a_{1j}} \cdot t_j, \dots, \frac{a_{ij}}{a_{1j}} \cdot t_j, \dots, \frac{a_{pj}}{a_{1j}} \cdot t_j \right)$, y por construcción, cada columna de W resulta ser:

$$w^{[j]} = \left(t_j, \frac{a_{2j}}{a_{1j}} \cdot t_j, \dots, \frac{a_{ij}}{a_{1j}} \cdot t_j, \dots, \frac{a_{pj}}{a_{1j}} \cdot t_j \right)^T \quad (2.51)$$

o, cada elemento de W sería:

$$w_{ij} = \frac{a_{ij}}{a_{1j}} \cdot t_j \quad (2.52)$$

de forma que se verificará:

$$A = W \cdot D \quad (2.53)$$

siendo:

$$d_{ii} = \frac{a_{1i}}{t_i} \quad (2.54)$$

y según la Definición 2.1, W es similar a A y las señales reconstruidas, y_i , se relacionan con las fuentes originales según:

$$y_i = d_{ii} \cdot s_i = \frac{a_{1i}}{t_i} \cdot s_i \quad (2.55)$$

Esta observación es de gran importancia ya que permite abstraer el orden de las aristas y su origen en el espacio de las fuentes.

Obsérvese que el orden de ubicación de los p vectores como columnas de W es arbitrario ya que ello equivale a una permutación en las columnas iniciales de A , y, según se indicó en la Sec.x.xx la matriz W encontrada sigue siendo válida.

A continuación se da otra forma de demostrar la *Propiedad 2.6*.

Consideremos un conjunto de vectores ortogonales del espacio fuente, con las siguientes coordenadas:

$$\mathbf{u}_j = (c_j \cdot \delta_{ij})^T \quad \mathbf{u}_j \in \mathfrak{R}^p \quad \mathbf{u}_j = (u_{ij})^T \quad i, j \in \{1, 2, \dots, p\} \quad (2.56)$$

Las imágenes de estos vectores en el espacio de observación según (2.2) son:

$$\mathbf{w}_j = A \cdot \mathbf{u}_j \quad \mathbf{w}_j \in \mathfrak{R}^p \quad \mathbf{w}_j = (w_{ij})^T \quad i, j \in \{1, 2, \dots, p\} \quad (2.57)$$

Es decir, las componentes de los vectores imagen $w_j = (w_{ij})$ son:

$$w_{ij} = \sum_{k=1}^p a_{ik} \cdot u_{kj} = a_{ij} \cdot c_j \quad (2.58)$$

Los vectores imagen w_j de los vectores ortogonales que corresponden al espacio S pueden considerarse como columnas de una matriz $W \in \mathfrak{R}^{p \times p}$, de componentes w_{ij} :

$$W = (w_{ij}) = A \cdot D_c \quad \text{donde} \quad D_c = (c_i \cdot \delta_{ij}) \quad (2.59)$$

Es decir, de (2.10), los vectores imagen de la base del espacio S forman una matriz W , similar a A , con $P=I$ y $D = D_c^{-1}$. El problema, por consiguiente, se limita a localizar esas imágenes.

Propiedad 2.7:

Puede utilizarse como vértice de referencia uno cualquiera de ellos.

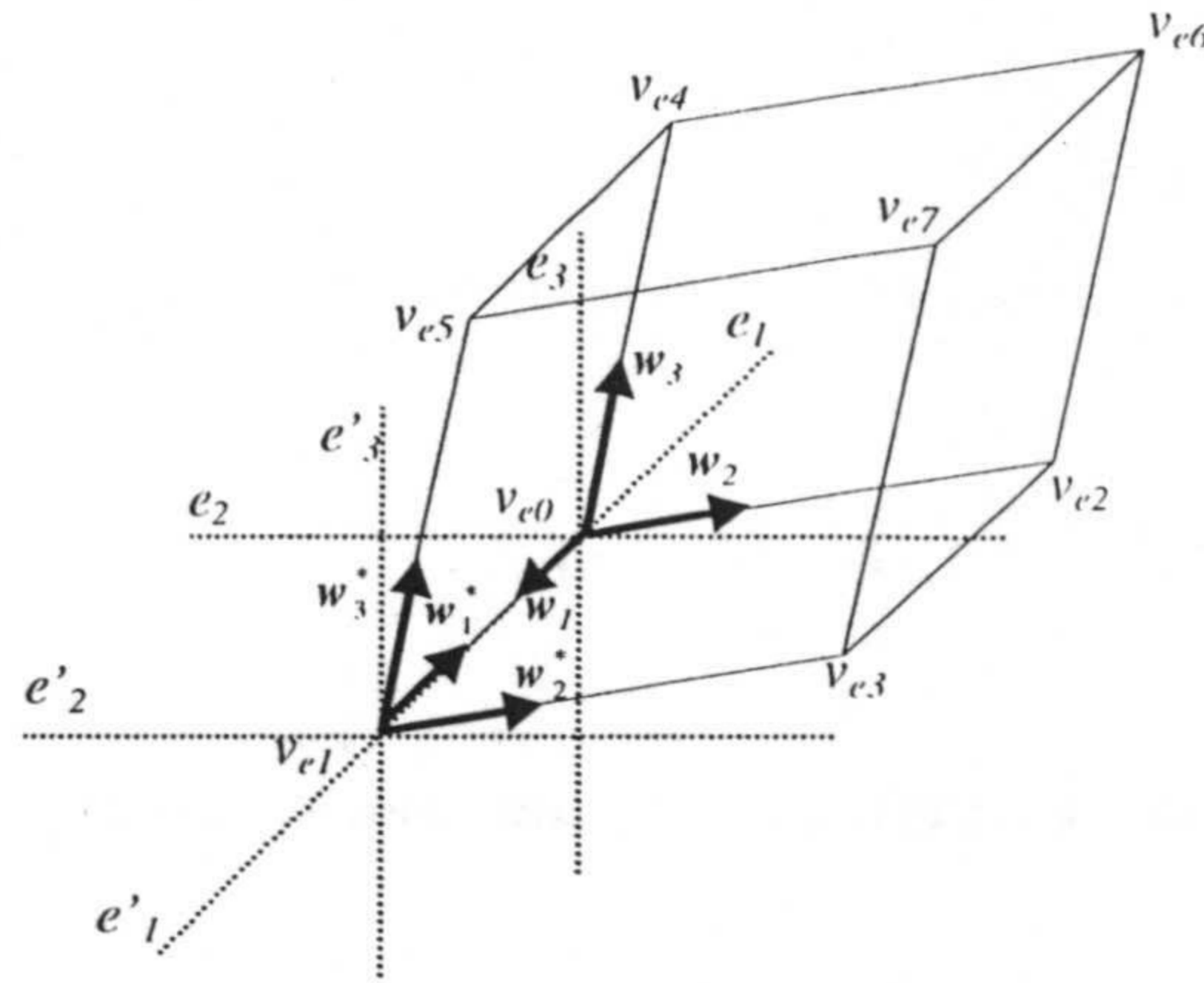


Figura 2.4. Efecto del cambio de vértice de referencia en los vectores w_j

Esta propiedad resulta evidente si se tiene en cuenta que p vectores arista incidentes en cualquier vértice son paralelos a cualquiera otros p vectores arista incidentes en otro vértice, verificándose siempre que:

$$w_{ij} = (c_i \cdot a_{ji}) \quad (2.60)$$

Por ejemplo, en la Figura 2.4: $c_1^* = -c_1$; $c_2^* = c_2$ y $c_3^* = c_3$. Es decir, el vértice que se considere como origen sólo influye en el signo de c_i .

Por lo tanto el vértice que se tome como referencia es irrelevante ya que con cualquier conjunto de p vectores arista incidentes en él se puede construir una matriz W equivalente a A . La diferencia entre las matrices W obtenidas utilizando como referencia uno u otro vértice se diferenciarán en una permutación de columnas y en factores de proporcionalidad de cada columna representables analíticamente por medio de una matriz diagonal. Así, si W es la matriz obtenida con vectores arista incidentes en un vértice, y si denominamos W^* a la matriz obtenida con vectores arista incidentes en otro vértice, se verificará que:

$$W^* = W \cdot Z \cdot D \quad (2.61)$$

Para comprender mejor la propiedad anterior considérese la Figura 2.4, que representa un posible espacio de observaciones para $p=3$. Supóngase que se toma como origen v_{e0} .

Supóngase también que las ecuaciones de las rectas que pasan por las aristas incidentes en v_{e0} (2.49) son:

$$\begin{aligned}
 \text{arista } [0;1]: \quad & \frac{e_1}{a_{11}} = \frac{e_2}{a_{21}} = \frac{e_3}{a_{31}} \\
 \text{arista } [0;2]: \quad & \frac{e_1}{a_{12}} = \frac{e_2}{a_{22}} = \frac{e_3}{a_{32}} \\
 \text{arista } [0;4]: \quad & \frac{e_1}{a_{13}} = \frac{e_2}{a_{23}} = \frac{e_3}{a_{33}}
 \end{aligned} \tag{2.62}$$

de (2.51) podemos expresar los vectores w_j de la siguiente forma:

$$\begin{aligned}
 w_1 &= \left(t_1, \frac{a_{21}}{a_{11}} \cdot t_1, \frac{a_{31}}{a_{11}} \cdot t_1 \right) \\
 w_2 &= \left(t_2, \frac{a_{22}}{a_{12}} \cdot t_2, \frac{a_{32}}{a_{12}} \cdot t_2 \right) \\
 w_3 &= \left(t_3, \frac{a_{23}}{a_{13}} \cdot t_3, \frac{a_{33}}{a_{13}} \cdot t_3 \right)
 \end{aligned} \tag{2.63}$$

llamando $c_j \stackrel{\text{def}}{=} \frac{a_{1j}}{t_j}$ y teniendo en cuenta (2.62) podemos expresar tres vectores arista cualesquiera incidentes en ese vértice así:

$$\begin{aligned}
 \text{vector en arista } [0;1]: \quad & w_1 = (c_1 \cdot a_{11}, c_1 \cdot a_{21}, c_1 \cdot a_{31}) = c_1 \cdot (a_{11}, a_{21}, a_{31}) \\
 \text{vector en arista } [0;2]: \quad & w_2 = (c_2 \cdot a_{12}, c_2 \cdot a_{22}, c_2 \cdot a_{32}) = c_2 \cdot (a_{12}, a_{22}, a_{32}) \\
 \text{vector en arista } [0;4]: \quad & w_3 = (c_3 \cdot a_{13}, c_3 \cdot a_{23}, c_3 \cdot a_{33}) = c_3 \cdot (a_{13}, a_{23}, a_{33})
 \end{aligned} \tag{2.64}$$

Si en lugar de v_{e0} se hubiese tomado como origen el vértice v_{e1} , debido al paralelismo entre las aristas incidentes en dichos vértices, las ecuaciones de las rectas que pasan por estas aristas serán:

$$\begin{aligned}
\text{arista } [1;0]: \quad \frac{e_1^*}{a_{12}} &= \frac{e_2^*}{a_{22}} = \frac{e_3^*}{a_{32}} \\
\text{arista } [1;3]: \quad \frac{e_1^*}{a_{11}} &= \frac{e_2^*}{a_{21}} = \frac{e_3^*}{a_{31}} \\
\text{arista } [1;5]: \quad \frac{e_1^*}{a_{13}} &= \frac{e_2^*}{a_{23}} = \frac{e_3^*}{a_{33}}
\end{aligned} \tag{2.65}$$

y las coordenadas de 3 puntos genéricos de los vectores aristas incidentes en el nuevo origen, v_{el} , serán:

$$\begin{aligned}
\text{vector en arista } [1;0]: \quad \mathbf{w}_1^* &= c_1^* \cdot (a_{12}, a_{22}, a_{32}) \\
\text{vector en arista } [1;3]: \quad \mathbf{w}_2^* &= c_2^* \cdot (a_{11}, a_{21}, a_{31}) \\
\text{vector en arista } [1;5]: \quad \mathbf{w}_3^* &= c_3^* \cdot (a_{13}, a_{23}, a_{33})
\end{aligned} \tag{2.66}$$

Con los 3 vectores (2.66) se puede construir la nueva matriz \mathbf{W}^* , verificándose:

$$\begin{aligned}
\mathbf{W}^* &= \begin{pmatrix} a_{12} & a_{11} & a_{13} \\ a_{22} & a_{21} & a_{23} \\ a_{32} & a_{31} & a_{33} \end{pmatrix} \cdot \begin{pmatrix} c_1^* & 0 & 0 \\ 0 & c_2^* & 0 \\ 0 & 0 & c_3^* \end{pmatrix} = \\
&= \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix} \cdot \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} c_1^* & 0 & 0 \\ 0 & c_2^* & 0 \\ 0 & 0 & c_3^* \end{pmatrix} = \mathbf{A} \cdot \mathbf{Z} \cdot \mathbf{D}
\end{aligned} \tag{2.67}$$

Es decir, según la Definición 2.1, se obtiene una matriz \mathbf{W}^* equivalente a \mathbf{A} .

Las *Propiedades* 2.6 y 2.7 son de extraordinaria importancia práctica para poder obtener una matriz \mathbf{W} similar a \mathbf{A} , ya que estas propiedades permiten hacer una abstracción completa del espacio de las fuentes. En efecto:

- a) Hacen innecesario considerar el orden de las aristas, y
- b) Posibilitan tener como referencia cualquier vértice del hiperparalelepípedo del espacio de observaciones.

Hasta ahora para demostrar las distintas propiedades, referenciábamos los vértices del espacio E indicando el vértice del espacio S de que eran imágenes (notación $e^{[vsj]}$, vértice del espacio E imagen del vértice v_{sj}). A partir de ahora, se puede omitir la referencia al espacio S, tanto para los vértices como para las aristas. En consecuencia se usará otra notación, v_{ej} , que sólo considera el espacio E.

Este paso, que implica trabajar sólo en el espacio E, es imprescindible en el campo de separación *ciega* de fuentes, ya que se supone que el espacio de las fuentes (o espacio S) es totalmente desconocido.

2.4 OBTENCIÓN DE LOS VÉRTICES DEL HIPERPARALELEPÍPEDO DEL ESPACIO DE OBSERVACIONES.

Los algoritmos que se describirán en el Capítulo 4 necesitan referir los vectores de mezclas observados a un vértice, v_{ej} . Para simplificar la notación en esta sección los vértices de mezclas se denotan con un único subíndice, $j=1, \dots, 2^p$, omitiéndose el subíndice e , ya que sólo se va a hacer referencia al espacio E de observaciones. Para poder referir los vectores de observación a uno de los vértices, se necesitan determinar las coordenadas de uno de ellos. Obviamente como es un proceso de separación ciega de fuentes, la determinación de uno de los vértices debe hacerse tan solo a partir de los vectores de observación.

El problema se reduce a seleccionar (o estimar), de entre los puntos $e(t)$ que se están detectando, aquellos que sean los vértices del hiperparalelepípedo.

Por comodidad, en primer lugar se analiza el caso $p=2$. Los puntos de mezcla que se detectan estarán incluidos dentro del paralelepípedo determinado por los vértices v_0, v_1, v_3, v_2 (Figura 2.5). Los vértices (v_0, v_1, v_2 y v_3) se pueden obtener algorítmicamente imponiendo las siguientes condiciones:

$$\begin{aligned}
 v_0 &= (e_1^{[0]}, e_2^{[0]}) & si & \quad -e_1^{[0]} - e_2^{[0]} = \max(-e_1(t) - e_2(t)) \equiv k_0 & \quad \forall t \\
 v_1 &= (e_1^{[1]}, e_2^{[1]}) & si & \quad +e_1^{[1]} - e_2^{[1]} = \max(+e_1(t) - e_2(t)) \equiv k_1 & \quad \forall t \\
 v_2 &= (e_1^{[2]}, e_2^{[2]}) & si & \quad -e_1^{[2]} + e_2^{[2]} = \max(-e_1(t) + e_2(t)) \equiv k_2 & \quad \forall t \\
 v_3 &= (e_1^{[3]}, e_2^{[3]}) & si & \quad +e_1^{[3]} + e_2^{[3]} = \max(+e_1(t) + e_2(t)) \equiv k_3 & \quad \forall t
 \end{aligned} \tag{2.68}$$

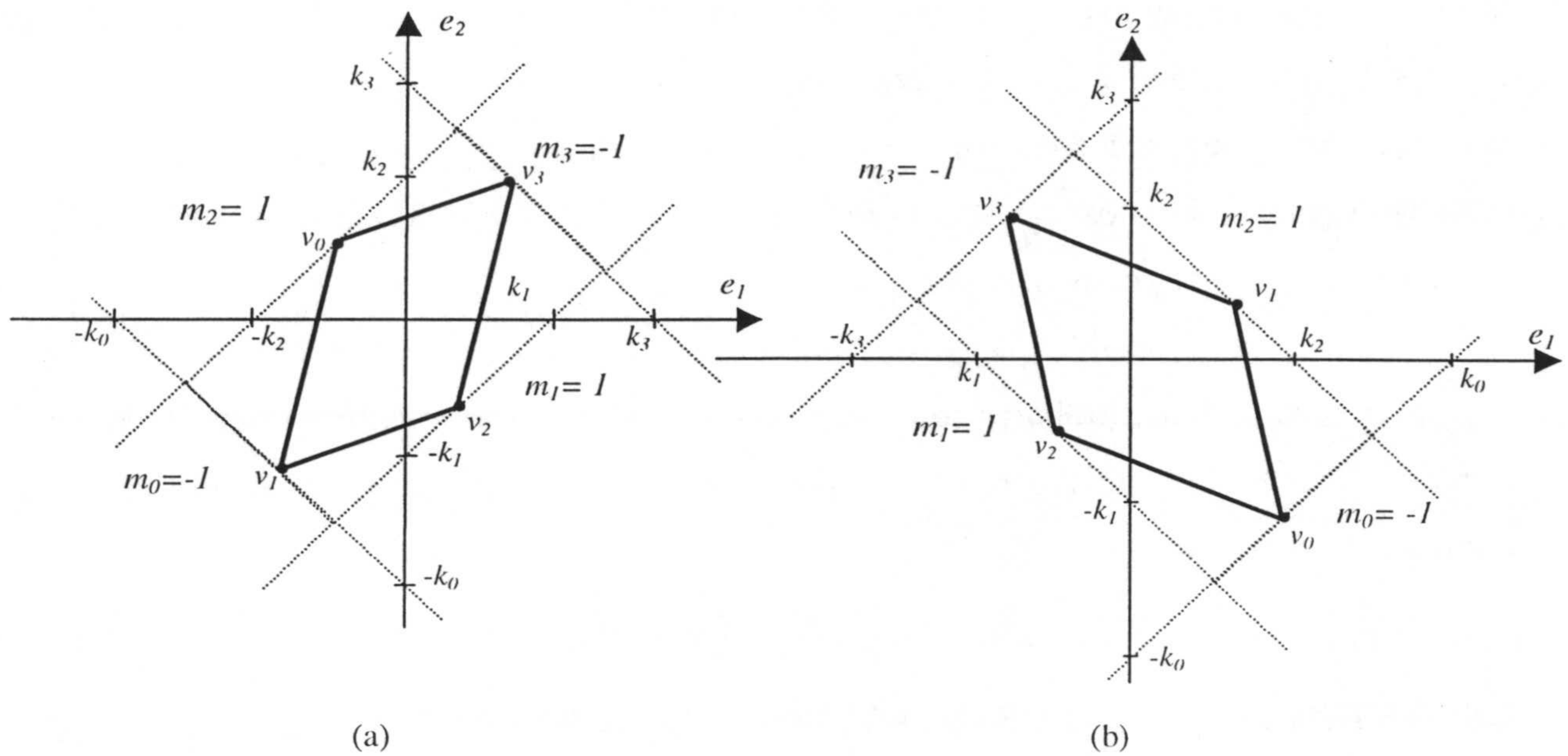


Figura 2.5. Obtención de los vértices a partir de los vectores del espacio E .

Es posible determinar el lugar geométrico de los puntos que pueden ser vértices. Estos lugares, en \mathfrak{R}^2 , son rectas. En efecto, llamando k_0, k_1, k_2 y k_3 a los valores de los máximos correspondientes a cada vértice, las coordenadas (s_1, s_2) de los posibles valores de mezcla cumplirán las siguientes condiciones:

$$\begin{aligned}
 \text{lugar } v_0 & \quad -e_1 - e_2 \leq k_0 \Rightarrow e_2 \geq -e_1 - k_0; \quad m_0 = -1 \\
 \text{lugar } v_1 & \quad +e_1 - e_2 \leq k_1 \Rightarrow e_2 \geq +e_1 - k_1; \quad m_1 = +1 \\
 \text{lugar } v_2 & \quad -e_1 + e_2 \leq k_2 \Rightarrow e_2 \geq +e_1 + k_2; \quad m_2 = +1 \\
 \text{lugar } v_3 & \quad +e_1 + e_2 \leq k_3 \Rightarrow e_2 \geq -e_1 + k_3; \quad m_3 = -1
 \end{aligned}
 \tag{2.69}$$

Obsérvese, por tanto, que los vértices pueden estar en las rectas determinadas por las ecuaciones (2.69) y que se representan en la Figura 2.5b con trazo punteado y en la Figura 2.6 rotuladas con los números 0, 1, 2 y 3. Estas rectas tienen de pendientes $m_0=-1, m_1=1, m_2=-1, m_3=1$, respectivamente, y los cortes con los ejes se producen en los puntos $\pm k_0, \pm k_1, \pm k_2$ y $\pm k_3$, respectivamente.

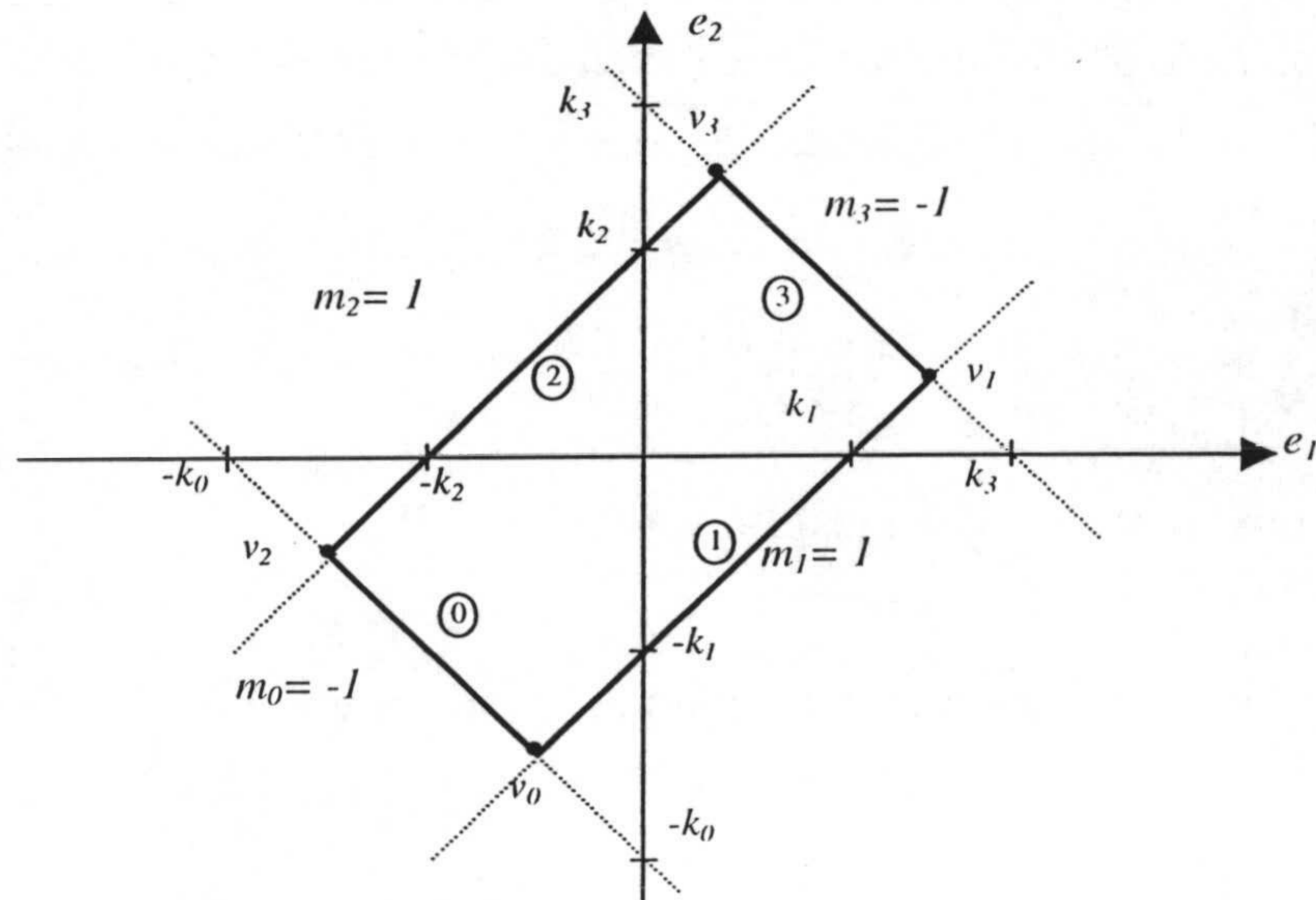


Figura 2.6. Lugares geométricos de los puntos que pueden ser vértices

Puede plantearse el problema de que más de 1 punto satisfaga la condición de vértice dada por (2.69). Éste sería el caso que se muestra en la Figura 2.6, que correspondería a la siguiente mezcla:

$$\begin{aligned} e_1 &= 0.4 \cdot s_1 - 0.6 \cdot s_2 \\ e_2 &= 0.4 \cdot s_1 + 0.6 \cdot s_2 \end{aligned} \quad (2.70)$$

donde $k_0 = k_3 = 4$ y $k_1 = k_2 = 6$.

En estas condiciones cualquier punto de la recta $[v_0; v_1]$ cumple la condición para ser vértice v_0 , y similarmente ocurre con los puntos de las rectas $[v_1; v_3]$, $[v_3; v_2]$ y $[v_2; v_1]$ que pueden ser vértices v_1 , v_3 y v_2 , respectivamente.

Se pueden plantear otras situaciones como la correspondiente a la siguiente mezcla:

$$\begin{aligned} e_1 &= 0.8 \cdot s_1 + 0.1 \cdot s_2 \\ e_2 &= -0.8 \cdot s_1 + 0.3 \cdot s_2 \end{aligned} \quad \text{con} \quad \begin{aligned} -5 &\leq s_1 \leq 5 \\ -5 &\leq s_2 \leq 5 \end{aligned} \quad (2.71)$$

En este caso los vértices según (2.68) serían:

- Vértice $v_0 \rightarrow \max(-e_1(t) - e_2(t)) = \max(-0.4 \cdot s_2) = 2$

Y las coordenadas de dicho vértice, teniendo en cuenta que $s_2=-5$, serán:

$$e_1 = 0.8 \cdot s_1 - 0.5$$

$$e_2 = -0.8 \cdot s_1 - 1.5$$

- Vértice $v_1 \rightarrow \max(e_1(t) - e_2(t)) = \max(1.6 \cdot s_1 - 0.2 \cdot s_2) = 9$

Y las coordenadas de dicho vértice, teniendo en cuenta que $s_1=5$ y $s_2=-5$, serán:

$$e_1 = 3.5$$

$$e_2 = -5.5$$

- Vértice $v_2 \rightarrow \max(-e_1(t) + e_2(t)) = \max(-1.6 \cdot s_1 + 0.2 \cdot s_2) = 9$

Y las coordenadas de dicho vértice, teniendo en cuenta que $s_1=-5$ y $s_2=+5$, serán:

$$e_1 = -3.5$$

$$e_2 = 5.5$$

- Vértice $v_3 \rightarrow \max(e_1(t) + e_2(t)) = \max(0.4 \cdot s_2) = 2$

Y las coordenadas de dicho vértice, teniendo en cuenta que $s_2=5$, serán:

$$e_1 = 0.8 \cdot s_1 + 0.5$$

$$e_2 = -0.8 \cdot s_1 + 1.5$$

En la Figura 2.7 se representa el espacio de mezclas correspondiente a los vértices obtenidos.

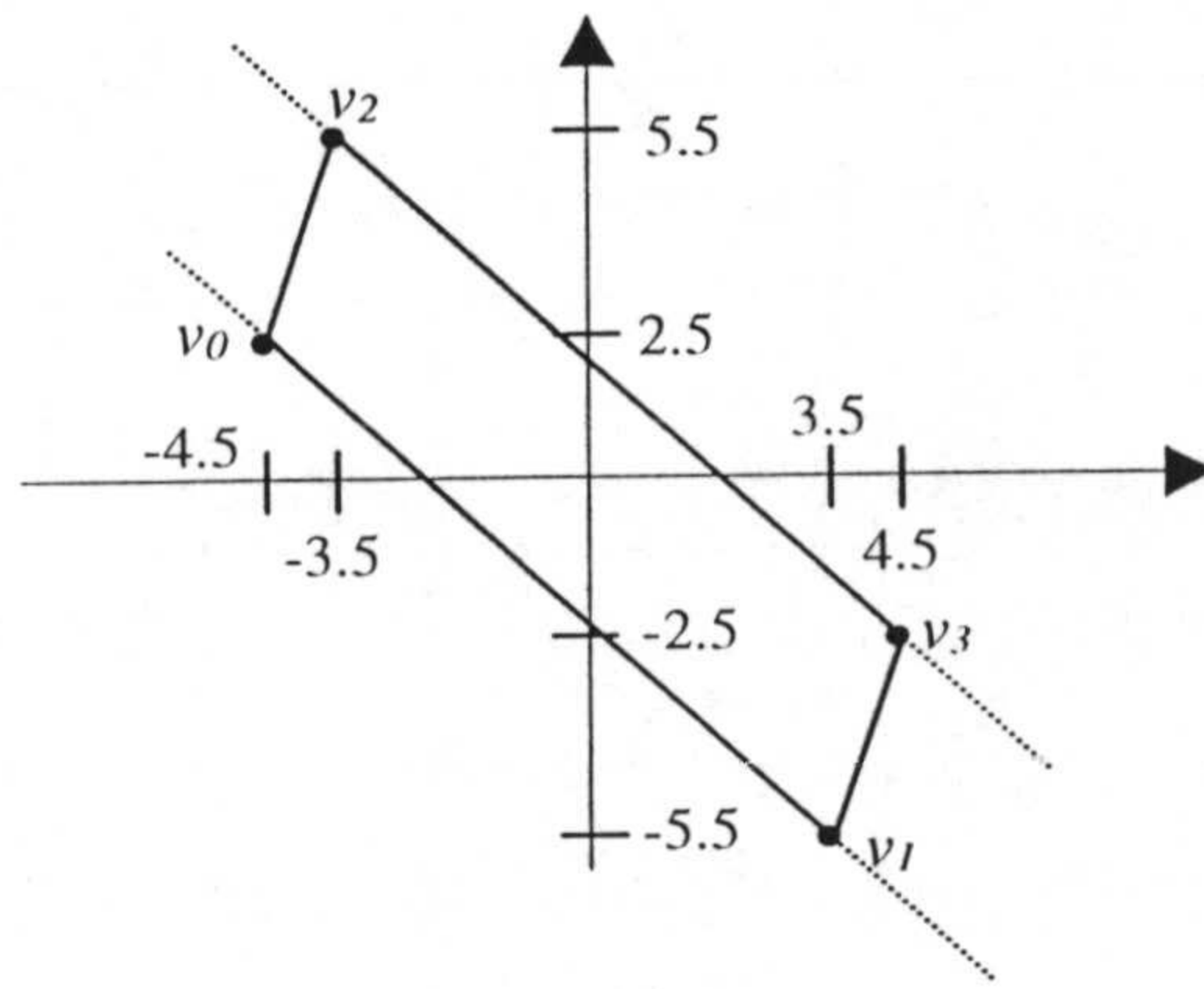


Figura 2.7. *Espacio de observaciones.*

Para evitar esta ambigüedad se impone una condición adicional al valor de la componente $e_2(t)$, de forma que un punto será vértice si verifica:

$$\begin{aligned}
 v_0 = (e_1^{[0]}, e_2^{[0]}) & \quad \text{si} \quad -e_1^{[0]} - e_2^{[0]} = \max(-e_1(t) - e_2(t)) \quad \text{y} \quad e_2^{[0]} = \min(e_2(t)) \quad \forall t \\
 v_1 = (e_1^{[1]}, e_2^{[1]}) & \quad \text{si} \quad +e_1^{[1]} - e_2^{[1]} = \max(+e_1(t) - e_2(t)) \quad \text{y} \quad e_2^{[1]} = \min(e_2(t)) \quad \forall t \\
 v_2 = (e_1^{[2]}, e_2^{[2]}) & \quad \text{si} \quad -e_1^{[2]} + e_2^{[2]} = \max(-e_1(t) + e_2(t)) \quad \text{y} \quad e_2^{[2]} = \max(e_2(t)) \quad \forall t \\
 v_3 = (e_1^{[3]}, e_2^{[3]}) & \quad \text{si} \quad +e_1^{[3]} + e_2^{[3]} = \max(+e_1(t) + e_2(t)) \quad \text{y} \quad e_2^{[3]} = \max(e_2(t)) \quad \forall t
 \end{aligned} \tag{2.72}$$

Los criterios anteriores se pueden generalizar para más dimensiones ($p > 2$). En efecto, un punto $(e_i^{[j]})$ es un vértice v_j , si verifica:

$$\begin{aligned}
 v_j = (e_i^{[j]}) \quad \text{si:} \\
 \sum_{i=1}^p (-1)^{[\Psi_{i-1, j+1}]} \cdot e_i^{[j]} = \max \sum_{i=1}^p (-1)^{[\Psi_{i-1, j+1}]} \cdot e_i(t); \quad \forall t; \quad \Psi_{i-1, j} \in \{0,1\}
 \end{aligned} \tag{2.73}$$

donde $\Psi_{i,j}$ es la cifra binaria de posición i correspondiente al número entero j representado en binario natural (ver Tabla 2.1, para $p=3$); es decir, se verifica:

$$j = \Psi_{p-1, j} \cdot 2^{p-1} + \Psi_{p-2, j} \cdot 2^{p-2} + \dots + \Psi_{0, j} \cdot 2^0 \tag{2.74}$$

Caso de que hubiese distintos puntos cumpliendo la condición de máximo, de entre ellos habría que seleccionar aquel que cumpliera la siguiente condición:

$$e_p^{[j]} = \max\left[(-1)^{\Psi_{ij}} \cdot e_p(t)\right] \quad \forall t \quad (2.75)$$

TABLA 2.2. Valores de Ψ_{ij} .

j	Ψ_{2j}	Ψ_{1j}	Ψ_{0j}
0	0	0	0
1	0	0	1
2	0	1	0
3	0	1	1
4	1	0	0
5	1	0	1
6	1	1	0
7	1	1	1

Los algoritmos anteriores se utilizarán en el Capítulo 4, al objeto de estimar la ubicación de los vértices. Es necesario encontrar las coordenadas de uno de los vértices, ya que hay que efectuar una traslación del hiperparalelepípedo (cambio de coordenadas), de forma tal que dicho vértice pase a ser el origen. Aunque sólo es necesario la utilización de uno de los vértices, la obtención de todos permite estimarlos con más precisión.

2.4.1 Relación entre los vértices.

Los vértices de un hiperparalelepípedo son puntos relacionados entre si. Conviene establecer analíticamente estas relaciones ya que pueden usarse para acelerar la determinación iterativa de los mismos. En efecto, los vectores de mezcla producidos por las fuentes, en general no estarán distribuidos uniformemente en el espacio E, con lo que la obtención de unos vértices puede ser más rápida (o probable) que la de los otros. Con ayuda de las relaciones que se indican en esta sección pueden obtenerse las coordenadas de un vértice conocidas las de los otros.

Geoméricamente es posible establecer las relaciones que se dan a continuación.

1. Los vértices definen aristas que son paralelas unas con otras.

La condición de paralelismo se puede describir analíticamente indicando que las aristas paralelas tienen igual pendiente. En concreto, para $p=2$, (Figura 2.8) se tiene que:

$$\begin{aligned}
& (v_0; v_2) \parallel (v_1; v_3): \\
& \frac{v_2(e_2) - v_0(e_2)}{v_2(e_1) - v_0(e_1)} = \frac{v_3(e_2) - v_1(e_2)}{v_3(e_1) - v_1(e_1)} \\
& (v_0; v_1) \parallel (v_2; v_3): \\
& \frac{v_1(e_2) - v_0(e_2)}{v_1(e_1) - v_0(e_1)} = \frac{v_3(e_2) - v_2(e_2)}{v_3(e_1) - v_2(e_1)}
\end{aligned}
\tag{2.76}$$

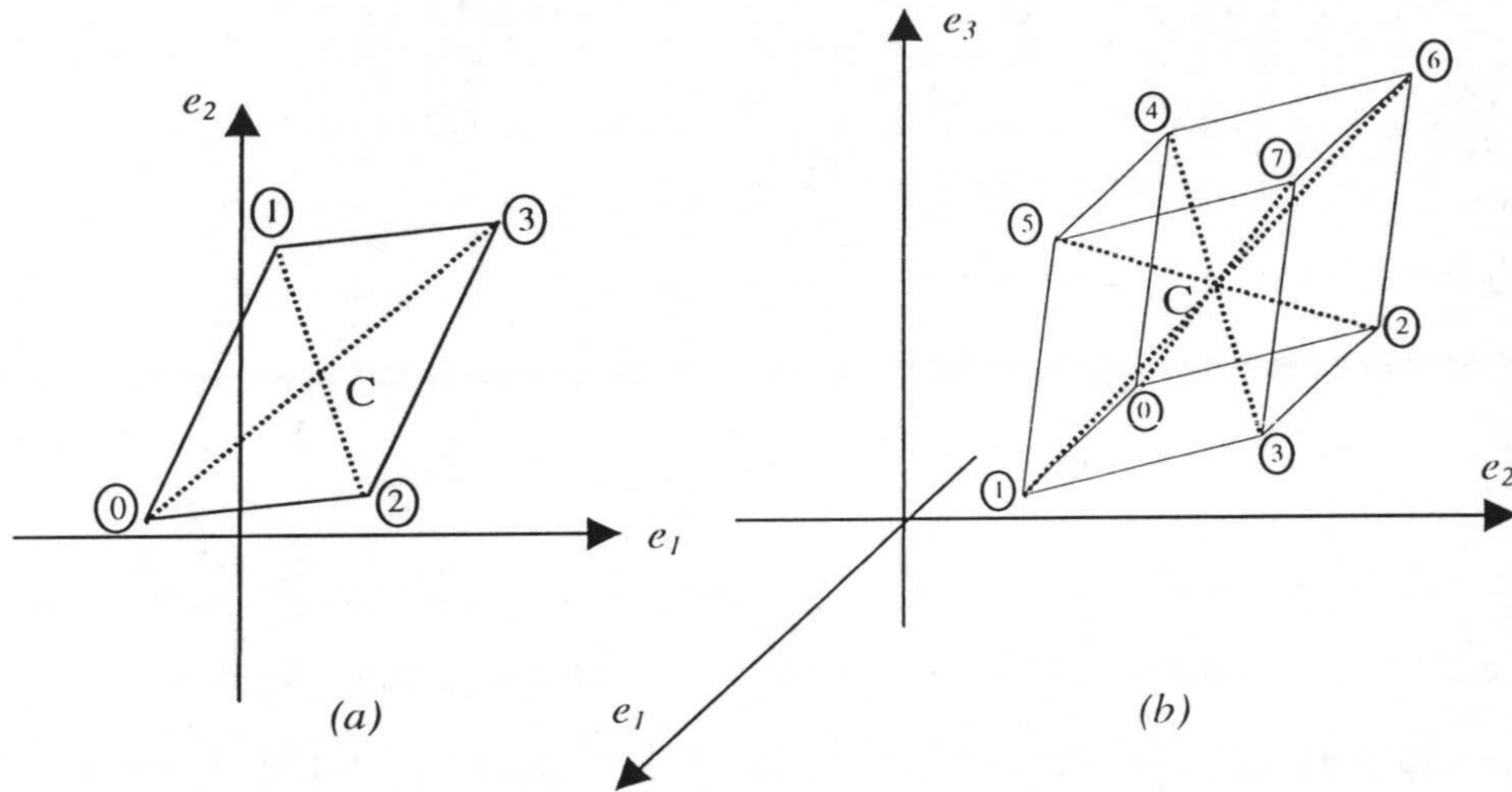


Figura 2.8. Relaciones entre los vértices. (a) $p=2$; (b) $p=3$.

En tres dimensiones ($p=3$, Figura 2.8b) se podrían establecer $3 \times 2 = 6$ relaciones, según se deduce de la Tabla 2.3, cada una de ellas con tres igualdades. Por ejemplo, las dos primeras relaciones serían:

$$\begin{aligned}
& (v_0; v_1) \parallel (v_2; v_3) \parallel (v_4; v_5) \parallel (v_6; v_7) :: \\
& \frac{v_1(e_2) - v_0(e_2)}{v_1(e_1) - v_0(e_1)} = \frac{v_3(e_2) - v_2(e_2)}{v_3(e_1) - v_2(e_1)} = \frac{v_5(e_2) - v_4(e_2)}{v_5(e_1) - v_4(e_1)} = \frac{v_7(e_2) - v_6(e_2)}{v_7(e_1) - v_6(e_1)} \\
& \frac{v_1(e_3) - v_0(e_3)}{v_0(e_1) - v_0(e_1)} = \frac{v_3(e_3) - v_2(e_3)}{v_3(e_1) - v_2(e_1)} = \frac{v_5(e_3) - v_4(e_3)}{v_5(e_1) - v_4(e_1)} = \frac{v_7(e_3) - v_6(e_3)}{v_7(e_1) - v_6(e_1)}
\end{aligned}
\tag{2.77}$$

TABLA 2.3. Aristas paralelas en un hiperparalelepípedo.

p=2	(0;1) (2;3)	(0;2) (1;3)		
p=3	(0;1) (2;3) (4;5) (6;7)	(0;4) (1;5) (2;6) (3;7)	(0;2) (1;3) (4;6) (5;7)	
p=4	(0;1) (2;3) (4;5) (6;7) (8;9) (10;11) (12;13) (14;15)	(0;2) (1;3) (4;6) (5;7) (8;10) (9;11) (12;14) (13;15)	(0;4) (1;5) (2;6) (3;7) (8;12) (9;13) (10;14) (11;15)	(0;8) (1;9) (2;10) (3;11) (4;12) (5;13) (6;14) (7;15)

En la Tabla 2.4 se indican las aristas que son paralelas y permiten obtener algorítmicamente las mismas. Esta tabla se ha obtenido partiendo de dos vértices adyacentes en el origen (primera fila de la Tabla 2.4), denotados en binario, que conformarían una arista original. Las paralelas a esta arista original se obtienen sin más que realizar todas las combinaciones posibles de los bits correspondientes a las posiciones en las que los vértices originales no cambian. En la Tabla 2.5 se aplica este procedimiento para $p=4$.

TABLA 2.4. Aristas paralelas en un hiperparalelepípedo para $p=4$ en notación binaria.			
(0000;0001)	(0000;0010)	(0000;0100)	(0000;1000)
(0010;0011)	(0001;0011)	(0001;0101)	(0001;1001)
(0100;0101)	(0100;0110)	(0010;0110)	(0010;1010)
(0110;0011)	(0101;0111)	(0011;0111)	(0011;1011)
(1000;1001)	(1100;1010)	(1000;1100)	(0100;1100)
(1010;1011)	(1001;1011)	(1001;1101)	(0101;1101)
(1100;1101)	(1100;1110)	(1010;1110)	(0110;1110)
(1110;1111)	(1101;1111)	(1011;1111)	(0111;1111)

2. Las diagonales entre vértices opuestos se cortan en un único punto equidistante de los vértices respectivos.

Considérese en primer lugar el caso $p=2$ (Figura 2.8a). Los vértices opuestos (Tabla 2.5) son: v_0 con v_3 y v_1 con v_2 . La condición para que se corten en el punto central puede expresarse analíticamente así:

$$\begin{aligned}
 &\text{componente } e_1 : \\
 &\frac{v_3(e_1) - v_0(e_1)}{2} + v_0(e_1) = \frac{v_1(e_1) - v_2(e_1)}{2} + v_2(e_1) \\
 &\text{componente } e_2 : \\
 &\frac{v_3(e_2) - v_0(e_2)}{2} + v_0(e_2) = \frac{v_1(e_2) - v_2(e_2)}{2} + v_2(e_2)
 \end{aligned} \tag{2.78}$$

La expresión (2.76) puede escribirse como:

$$\begin{aligned}
 v_3(e_1) + v_0(e_1) &= v_2(e_1) + v_1(e_1) \\
 v_3(e_2) + v_0(e_2) &= v_2(e_2) + v_1(e_2)
 \end{aligned} \tag{2.79}$$

La generalización para dimensiones mayores ($p>2$) es fácil. Para $p=3$ (Figura 2.8b) se tendría:

$$\begin{aligned}
 v_0(e_1) + v_7(e_1) &= v_1(e_1) + v_6(e_1) = v_2(e_1) + v_5(e_1) = v_3(e_1) + v_4(e_1) \\
 v_0(e_2) + v_7(e_2) &= v_1(e_2) + v_6(e_2) = v_2(e_2) + v_5(e_2) = v_3(e_2) + v_4(e_2) \\
 v_0(e_3) + v_7(e_3) &= v_1(e_3) + v_6(e_3) = v_2(e_3) + v_5(e_3) = v_3(e_3) + v_4(e_3)
 \end{aligned} \tag{2.80}$$

En general, para $p>3$, al aplicar las relaciones (2.80) el problema se reduce a determinar los índices de los vértices opuestos. La regla se puede obtener a partir de la Figura 2.8 y de la Tabla 2.5. Obsérvese que dado un punto, x , su opuesto es $[(2^p-x)-1]$.

TABLA 2.5. Vértices opuestos.

p=2	p=3	p=4
(0;3)	(0;7)	(0;15)
(1;2)	(1;6)	(1;14)
	(2;5)	(2;13)
	(3;4)	(3;12)
		(4;11)
		(5;10)
		(6;9)
		(7;8)

3. Los vértices definen aristas cuyas longitudes en aristas paralelas son iguales.

Esta propiedad resulta evidente ya que dos aristas paralelas están cortadas a su vez por rectas paralelas.

Para $p=2$, esta propiedad se puede expresar analíticamente así:

$$\begin{aligned} (v_0; v_2) \parallel (v_1; v_3) \text{ y } d(v_0, v_2) = d(v_1, v_3): \\ \sqrt{(v_2(e_1) - v_0(e_1))^2 + (v_2(e_2) - v_0(e_2))^2} = \sqrt{(v_3(e_1) - v_1(e_1))^2 + (v_3(e_2) - v_1(e_2))^2} \\ (v_0; v_1) \parallel (v_2; v_3) \text{ y } d(v_0, v_1) = d(v_2, v_3): \\ \sqrt{(v_1(e_1) - v_0(e_1))^2 + (v_1(e_2) - v_0(e_2))^2} = \sqrt{(v_3(e_1) - v_2(e_1))^2 + (v_3(e_2) - v_2(e_2))^2} \end{aligned} \quad (2.81)$$

Donde $d(v_i, v_j)$ es la distancia geométrica entre los vértices v_i y v_j . Siendo fácilmente generalizable para $p > 2$.

En este tercer caso la relación se establece entre aristas paralelas por lo que habrá que aplicar el algoritmo de determinación de paralelas visto en el caso 1 (Tabla 2.4).

Matemáticamente unas propiedades están relacionadas con otras. Desde un punto de vista práctico, lo más útil es utilizar la propiedad segunda ya que tanto las relaciones entre vértices (2.80) como el algoritmo de establecimiento de vértices opuestos son los más sencillos.

2.4.2 Ejemplo.

Para comprobar las propiedades anteriores se considera el hiperparalelepípedo siguiente, para $p=3$, cuyos vértices son (Figura 2.9):

$$\begin{aligned} v_0 &= (4,11,8) & v_4 &= (10,18,20) \\ v_1 &= (1,3,2) & v_5 &= (7,10,14) \\ v_2 &= (5,18,10) & v_6 &= (11,25,22) \\ v_3 &= (2,10,4) & & \end{aligned} \quad (2.82)$$

Conocidos los vértices $v_0, v_1, v_2, v_3, v_4, v_5,$ y v_6 , se puede obtener el vértice v_7 a partir de la expresión (2.80); es decir, desglosando las igualdades se tienen las siguientes seis expresiones para cada coordenada (e_1, e_2, e_3) :

$$\begin{aligned}
 v_0(e_1) + v_7(e_1) &= v_1(e_1) + v_6(e_1) \\
 v_0(e_1) + v_7(e_1) &= v_2(e_1) + v_5(e_1) \\
 v_0(e_1) + v_7(e_1) &= v_3(e_1) + v_4(e_1) \\
 v_1(e_1) + v_6(e_1) &= v_2(e_1) + v_5(e_1) \\
 v_1(e_1) + v_6(e_1) &= v_3(e_1) + v_4(e_1) \\
 v_2(e_1) + v_5(e_1) &= v_3(e_1) + v_4(e_1)
 \end{aligned}
 \tag{2.83}$$

Estas ecuaciones son linealmente dependientes, quedando al final el siguiente sistema de ecuaciones:

$$\begin{aligned}
 3v_0(e_1) + v_1(e_1) - v_2(e_1) - 3v_3(e_1) - 3v_4(e_1) - v_5(e_1) + v_6(e_1) + 3v_7(e_1) &= 0 \\
 3v_0(e_2) + v_1(e_2) - v_2(e_2) - 3v_3(e_2) - 3v_4(e_2) - v_5(e_2) + v_6(e_2) + 3v_7(e_2) &= 0 \\
 3v_0(e_3) + v_1(e_3) - v_2(e_3) - 3v_3(e_3) - 3v_4(e_3) - v_5(e_3) + v_6(e_3) + 3v_7(e_3) &= 0
 \end{aligned}
 \tag{2.84}$$

Sustituyendo los valores de los vértices dados en (2.82) y aplicando (2.84) se obtiene el siguiente valor para el vértice v_7 :

$$v_7 = (8, 17, 16)
 \tag{2.85}$$

que coincide con su valor real.

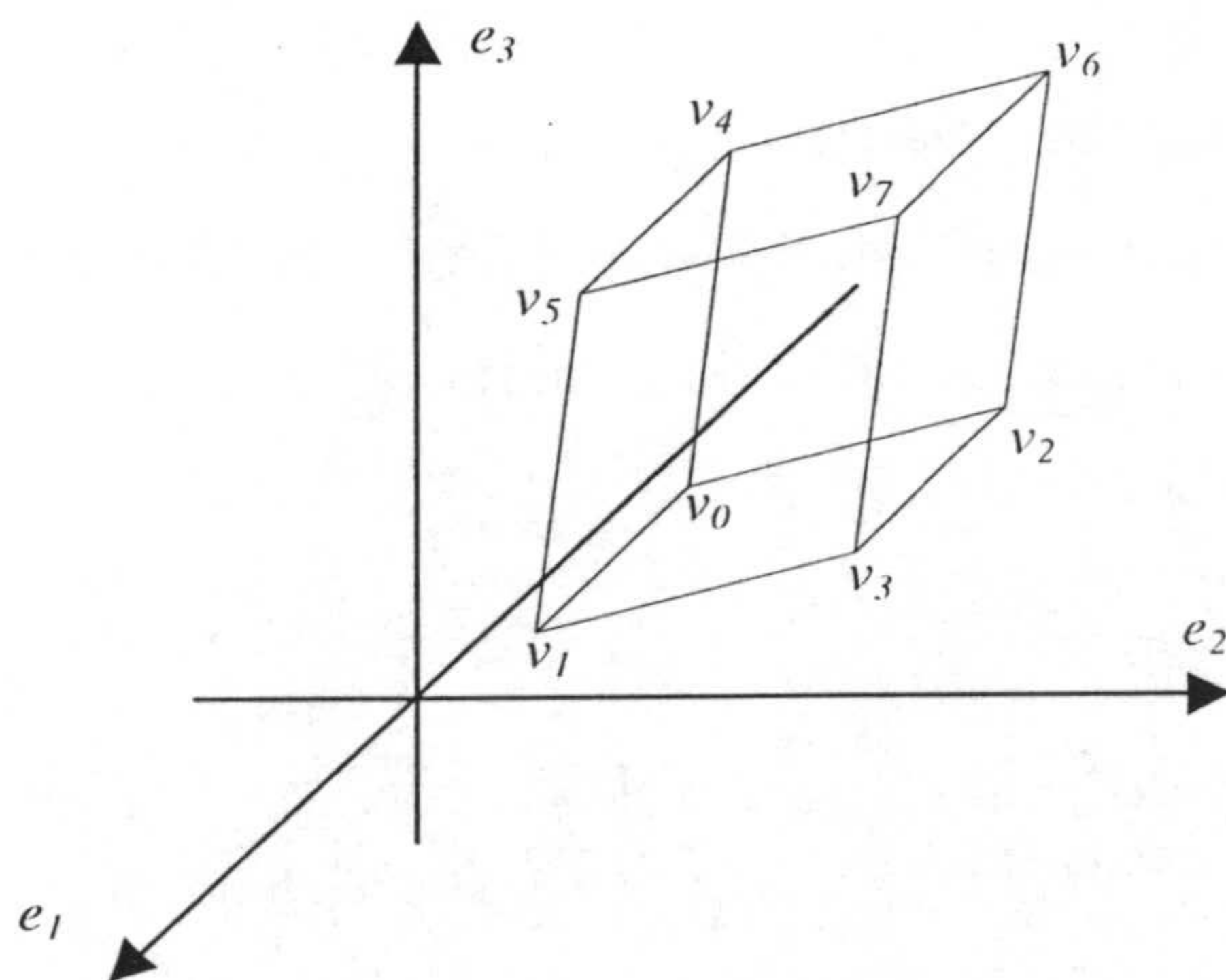


Figura 2.9. Hiperparalelepípedo del espacio de observaciones para $p=3$.

2.5 MEDIDAS DE LAS PRESTACIONES.

Según se ha indicado en las secciones anteriores, el objetivo de la resolución del problema de separación de fuentes es obtener una matriz W similar a A .

Los algoritmos de separación usualmente no obtienen una matriz W_0 exactamente similar a A , sino una serie de matrices $W(t)$ que, dentro de un proceso iterativo, se van sucesivamente aproximando cada vez más a W_0 . En ambos casos interesa medir el grado de aproximación de $W(t)$ a W_0 . Para ello se van a utilizar a lo largo de la memoria unos índices que medirán de forma objetiva la aproximación de $W(t)$ a W_0 (que en definitiva representa la calidad de la separación), o su variación en el tiempo dentro de un proceso iterativo que indicará el grado de convergencia del método utilizado para la separación.

Los índices que se van a usar son:

- Índice de prestaciones de Amari.
- Error cuadrático medio entre los elementos de A y W .
- Error cuadrático medio entre $y(t)$ y $s(t)$.
- Diafonía ("*Crosstalk*").

A continuación se definen estos índices.

2.5.1 Índice de prestaciones de Amari.

Amari, Cichocki y Yang han propuesto [AMA95, AMA96] un índice de prestaciones (que denominaremos PII), con el que se puede determinar la calidad de la separación de fuentes con un solo valor. Este índice se define como:

$$PII \stackrel{def}{=} \sum_{i=1}^p \left(\sum_{j=1}^p \frac{|p_{ij}|}{\max_k |p_{ik}|} - 1 \right) + \sum_{j=1}^p \left(\sum_{i=1}^p \frac{|p_{ij}|}{\max_k |p_{kj}|} - 1 \right) \quad (2.86)$$

donde $P = (p_{ij})$ es la matriz de prestaciones tal como se definió en (2.10). Cuanto menor sea el valor del índice, mejores son las prestaciones (W es más similar a A), y la separación es perfecta cuando $PII = 0$.

Obsérvese que según (2.10) si W es similar a A :

$$P = D \cdot Z \quad (2.87)$$

es decir, como se indica en la expresión (2.11) para $p=2$, P es una matriz en la que todos los elementos de cada columna y cada fila son 0 salvo uno. Precisamente la expresión (2.81) lo que hace es comprobar si se verifica esta condición. En efecto, supóngase que se tienen tres señales y la matriz Z es de la forma (2.12), entonces el valor de PII es:

$$PII = \frac{1}{\varepsilon_{11} \cdot c_1} (\varepsilon_{12} + \varepsilon_{13} + \varepsilon_{21} + \varepsilon_{31}) + \frac{1}{\varepsilon_{22} \cdot c_2} (\varepsilon_{23} + \varepsilon_{21} + \varepsilon_{12} + \varepsilon_{32}) + \frac{1}{\varepsilon_{33} \cdot c_3} (\varepsilon_{32} + \varepsilon_{31} + \varepsilon_{13} + \varepsilon_{23}) \quad (2.88)$$

De (2.83) se deduce que PI será más pequeño cuanto más se aproximen a cero los valores de ε_{ij} ($i \neq j$), en este último caso se tiene:

$$PII \rightarrow \frac{1}{\varepsilon_{11} \cdot c_1} \cdot 0 + \frac{1}{\varepsilon_{22} \cdot c_2} \cdot 0 + \frac{1}{\varepsilon_{33} \cdot c_3} \cdot 0 \rightarrow 0 \quad (2.89)$$

indicando que W es similar a A , como es el caso.

En otras palabras, durante el aprendizaje la matriz P tiende a una matriz permutación generalizada ($D \cdot Z$) en la que cada columna y cada fila tienen uno y sólo un elemento dominante. El índice PII estima o mide esta propiedad.

Para $p=2$ la expresión (2.81) se convierte en:

$$PII = \left[\frac{|p_{11}|}{\max_k |p_{1k}|} + \frac{|p_{12}|}{\max_k |p_{1k}|} - 1 \right] + \left[\frac{|p_{21}|}{\max_k |p_{2k}|} + \frac{|p_{22}|}{\max_k |p_{2k}|} - 1 \right] + \left[\frac{|p_{11}|}{\max_k |p_{k1}|} + \frac{|p_{21}|}{\max_k |p_{k1}|} - 1 \right] + \left[\frac{|p_{12}|}{\max_k |p_{k2}|} + \frac{|p_{22}|}{\max_k |p_{k2}|} - 1 \right] \quad (2.90)$$

donde k es el índice del máximo de cada fila y columna.

Por ejemplo, si obtenemos la siguiente matriz \mathbf{P} :

$$\mathbf{P} = \begin{bmatrix} 0 & -3 \\ 7 & 0 \end{bmatrix} \quad (2.91)$$

querría decir que W es similar a A (sólo hay un elemento en cada fila y cada columna distinto de 0), y el índice de prestaciones aplicando la expresión (2.85) resulta ser:

$$P_{11} = \frac{0}{|-3|} + \frac{|-3|}{|-3|} - 1 + \frac{7}{7} + \frac{0}{7} - 1 + \frac{0}{7} + \frac{7}{7} - 1 + \frac{|-3|}{|-3|} + \frac{0}{|-3|} - 1 = 0 \quad (2.92)$$

Los valores $p_{12}=-3$ y $p_{21}=7$ significan que las señales originales se recuperarán amplificadas por 3 y 7, respectivamente, y la primera estará invertida.

2.5.2 Error cuadrático medio.

Otros índices que se pueden usar para medir la calidad del proceso de separación de fuentes se basan en el concepto de error cuadrático medio.

Si se tienen dos señales, o funciones, $x(t)$ e $y(t)$, donde $x(t)$ es una estimación de $x_0(t)$, se define el error cuadrático como:

$$EC(t) = \sqrt{[x_0(t) - x(t)]^2} \quad (2.93)$$

Obsérvese que el error cuadrático es sencillamente el error absoluto, $x(t) - x_0(t)$; pero para que el error se contabilice sin tener en cuenta el signo, ($x_0^{(t)} > x(t)$ o $x_0^{(t)} < x(t)$) con objeto de que los errores negativos no compensen a los positivos, el error absoluto se eleva

al cuadrado. Posteriormente se extrae la raíz cuadrada para que el valor del error obtenido sea del orden de magnitud de las señales originales.

Basándose en el error cuadrático se pueden definir dos índices de prestaciones. Uno considerando la matriz original A y la estimación de la misma, W , y otro que compara las señales originales, $s(t)$, con las señales reconstruidas, $y(t)$. A continuación se analizan estos dos índices.

2.5.2.1. Error cuadrático medio entre los elementos de W y A .

El concepto de error cuadrático medio definido por la expresión (2.93) no es sólo aplicable a funciones, sino también a datos o valores individuales. Si por ejemplo el valor z es una aproximación del valor z_0 , el error cuadrático medio de esta estimación será:

$$ECM(t) = \sqrt{[z_0 - z]^2} \quad (2.94)$$

Para hallar el error cuadrático medio entre los elementos de W y A hay que calcular las diferencias entre los elementos de $w_{ij} - a_{ij}$ ($i, j=1, \dots, p$). Se plantea un doble problema debido a que:

- 1) las columnas de W están desordenadas respecto a las de A , y
- 2) las columnas j de W están multiplicadas por una constante, c_j , respecto a las de A .

Esto provoca que las diferencias $w_{ij} - a_{ij}$ no sean significativas ni útiles para obtener una medida de error ya que habría que obtener dicho error con W ordenada y un factor de escala, en ambos casos respecto a A .

El problema de factor de escala de las columnas se puede paliar sin más que normalizar a 1 las columnas tanto de A como de W antes de efectuar la resta. A partir de $A = (a_{ij})$ y $W = (w_{ij})$ se obtienen otras matrices $A^* = (a^*_{ij})$ y $W^* = (w^*_{ij})$ respectivamente, tales que:

$$a_{ij}^* = \frac{a_{ij}}{\sum_{j=1}^p a_{ij}} \quad w_{ij}^* = \frac{w_{ij}}{\sum_{j=1}^p w_{ij}} \quad (2.95)$$

$$\forall i, j \in \{0, 1, \dots, p\}$$

Una alternativa, que es la que se utilizará en el Capítulo 5, es dividir los elementos de cada columna de la matriz por el elemento de la diagonal principal, de forma que la matriz resultante tendrá todos los elementos de la diagonal igual a 1. La matriz \mathbf{A} así obtenida, la denominaremos *forma canónica de A* y la representaremos por $\mathbf{A}_d = (a_{dij})$. En consecuencia, se verificará:

$$a_{dij} = \frac{a_{ij}}{a_{ii}} \quad \forall i, j \in \{1, \dots, p\} \quad (2.96)$$

o:

$$\mathbf{A}_d = \begin{pmatrix} 1 & \frac{a_{12}}{a_{11}} & \dots & \frac{a_{1p}}{a_{11}} \\ \frac{a_{21}}{a_{11}} & 1 & \dots & \frac{a_{2p}}{a_{11}} \\ \vdots & \vdots & \dots & \vdots \\ \frac{a_{p1}}{a_{11}} & \frac{a_{p2}}{a_{11}} & \dots & 1 \end{pmatrix} \quad (2.97)$$

Por otra parte la reordenación de columnas se puede resolver fácilmente cuando se consideran problemas sintéticos, por ejemplo, imponiendo la condición de que en la matriz \mathbf{A} sean máximos los elementos de la diagonal principal en sus columnas y filas respectivas, es decir:

$$a_{ii} > a_{ij} \quad \forall i, j \in \{1, 2, \dots, p\}, i \neq j \quad (2.98)$$

En un caso real, esto equivaldría a suponer que cada uno de los sensores, y_i , es más sensible a una fuente distinta, s_i . Esto, físicamente quiere decir que los sensores se encuentran adecuadamente separados uno de otro.

Si se cumple esta condición, se puede reordenar la matriz obtenida, W , (similar a la matriz A) imponiendo la misma condición (2.89) a sus elementos; es decir, se reordena W de forma tal que se obtenga una matriz W^* tal que:

$$w_{ii}^* > w_{ij}^* \quad \forall i, j \in \{1, 2, \dots, p\}, i \neq j \quad (2.99)$$

Denotando con W^* la matriz con columnas normalizadas y reordenada, y con A^* la matriz A con columnas normalizadas, se puede hacer la comprobación directa entre sus elementos respectivos para evaluar el resultado de la separación de fuentes. Si la separación es totalmente correcta los elementos de W_0^* deben ser exactamente iguales a los de A^* ($W_0^* = A^*$), y ahora sí se puede utilizar como medida de la calidad del error cuadrático medio (que denominamos ECM):

$$ECM(t) \stackrel{def}{=} \frac{1}{p^2} \sqrt{\sum_{i=1}^p \sum_{j=1}^p (a_{ij}^* - w_{ij}^*)^2} \quad (2.100)$$

Teniendo en cuenta que p es una constante y que la raíz cuadrada es una función monótona, se puede utilizar como índice de prestaciones el siguiente, que es más sencillo que el ECM :

$$PI2(t) = \sum_{i=1}^p \sum_{j=1}^p [a_{ij} - w_{ij}^*(t)]^2 \quad (2.101)$$

Si dentro de un proceso iterativo el método de separación converge se verificará:

$$\lim_{t \rightarrow \infty} PI2(t) = 0 \quad (2.102)$$

Para indicar la convergencia del algoritmo, resulta más significativo utilizar el valor medio, $PI2_v(N_t)$, del índice de prestaciones dentro de una ventana de tiempo deslizante definida entre $t=N_0$ y $t=N_t$, donde N_t es el número de muestra actual y N_0 es el número de muestra inicial de la ventana. La longitud de la ventana, N_v , (número de muestras) será:

$$N_v = N_t - N_0 \quad (2.103)$$

Con lo que el valor medio del índice de prestaciones dentro de la ventana en el instante $t = N_t$:

$$PI2_v(N_t) = \frac{1}{N_t} \cdot \sum_{t=N_0}^{N_t} PI2(t) \quad (2.104)$$

La expresión anterior se puede obtener de forma recursiva, suponiendo conocido $PI2_v(N_t - 1)$, de la siguiente manera:

$$PI2_v(N_t) = PI2_v(N_t - 1) - N_v \cdot PI2(N_t - 1 - N_v) + \frac{PI2(N_t)}{N_v} \quad (2.105)$$

Obsérvese que para evaluar iterativamente la expresión (2.101) es necesario memorizar todos los valores de $PI2$ dentro de la ventana en curso: $PI2(N_0)$, $PI2(N_0+1)$, ..., $PI2(N_t)$.

En el Capítulo 5, en vez de utilizar las matrices \mathbf{A} y \mathbf{W} normalizadas por columnas, se utilizan dichas matrices en su forma canónica, viniendo dado el error cuadrático medio ε , entre los elementos de ambas matrices, por:

$$\varepsilon = ECM(A_d, W_d) = \frac{1}{p^2} \sqrt{\sum_{i=1}^p \sum_{j=1}^p (a_{dij} - w_{dij})^2} \quad (2.106)$$

2.5.2.2. Error cuadrático medio entre las fuentes originales y las fuentes reconstruidas.

En este caso se trata de estimar la similitud entre las fuentes originales, s_i ($i=1, \dots, p$) y las señales reconstruidas, y_i ($i=1, \dots, p$).

Análogamente a lo que ocurría al tratar de comparar las matrices \mathbf{A} y \mathbf{W} , ahora se presentan dos problemas debido a las indeterminaciones admitidas de y_i frente a x_i :

- 1) El orden es distinto. Denotemos y_{i^*} a las señales y_i reordenadas con respecto a las fuentes originales.

2) Las señales originales multiplicadas por una constante, es decir:

$$y_{i^*}(t) = c_i \cdot s_i(t) \quad (2.107)$$

El primer problema se soluciona reordenando la matriz W en la forma que se ha descrito en la Sección 2.5.2.1. De esta forma se verificará:

$$y_{i^*}(t) = (W^*)^{-1} \cdot e(t) \quad (2.108)$$

El segundo problema se puede solucionar normalizando las señales y_{i^*} , s_i , dentro de los mismos límites. Esto se puede realizar como se describe a continuación.

Supongamos que s_{im} y s_{iM} son los valores mínimo y máximo respectivamente de s_i (2.1). Entonces podemos cambiar la escala de las señales y_{i^*} en concordancia con las de s_i aplicando la siguiente expresión:

$$y_i^* = \frac{y_{i^*} - y_{i^*}(\min)}{y_{i^*}(\max) - y_{i^*}(\min)} \cdot [s_{iM} - s_{im}] + s_{im} \quad (2.109)$$

de esta forma se podrán comparar, a efectos de medir el error, directamente las señales s_i e y_i^* (con y_i^* denotamos la señal recuperada correspondiente a s_i y normalizada también respecto a s_i).

Por ejemplo, supongamos que $s_{im} = -10$ y $s_{iM} = 10$. Caso de que $y_{i^*}(\max) = 5$ e $y_{i^*}(\min) = -5$, el valor correspondiente a $y_{i^*} = 2$ en la nueva escala será (Figura 2.10):

$$y_i^* = \frac{2 - (-5)}{5 - (-5)} \cdot [10 - (-10)] + (-10) = 4 \quad (2.110)$$

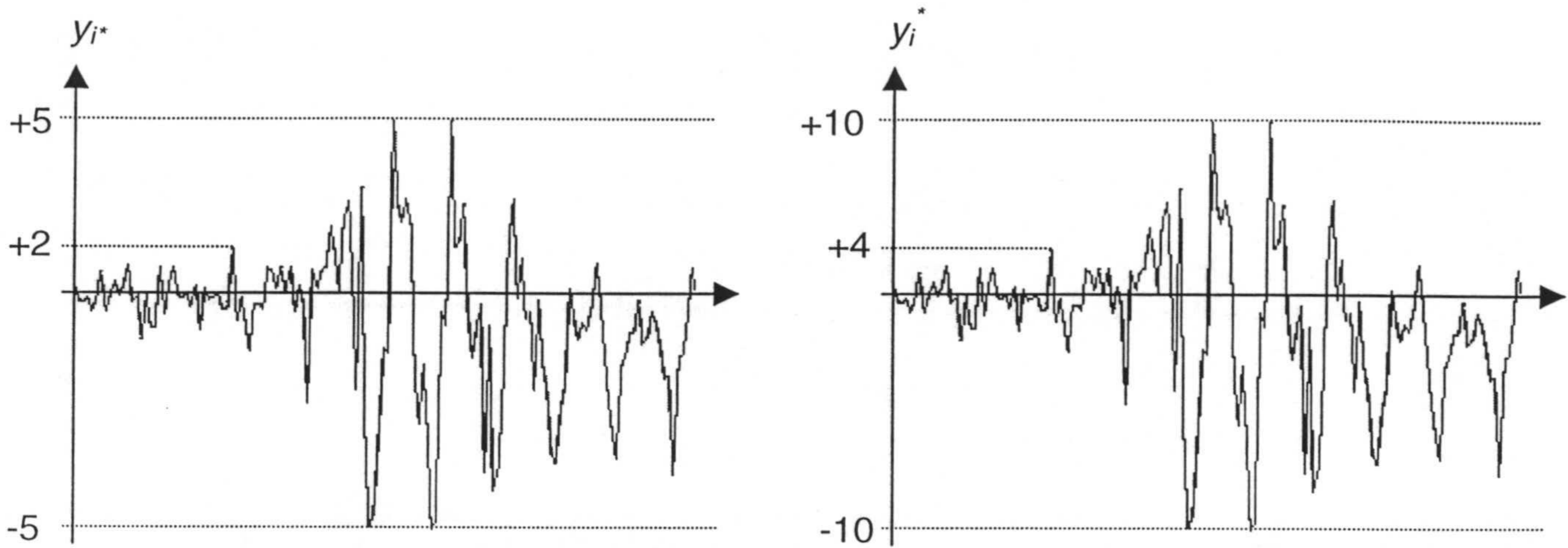


Figura 2.10. Ejemplo de cambio de escala.

El error cuadrático (RS, "Root Square") (2.93) en un instante de tiempo t será:

$$RS(t) = \sqrt{[s_i(t) - y_i^*(t)]^2} \quad (2.111)$$

Puede ocurrir que en un instante dado, t , dado el error sea grande y en el siguiente instante de tiempo sea pequeño. Una medida más precisa de la similitud entre $s_i(t)$ e $y_i^*(t)$ se obtiene si se efectúa la medida de los valores del error cuadrático dentro de una ventana temporal, v . Este valor, denominado *error cuadrático medio* (RMS, "Root Mean Square") viene dado por:

$$RMS_{i(v)}(N_t) = \frac{1}{N_v} \sum_{t=0}^{N_t} \sqrt{[s_i(t) - y_i^*(t)]^2} \quad (2.112)$$

Se puede obtener el error cuadrático medio con (2.109) donde se toma el inicio de la señal con $N_0=0$ y $N_v=N_t$.

La expresión (2.109) puede calcularse recursivamente de la forma:

$$RMS_{i(v)}(N_t) = RMS_{i(v)}(N_t - 1) - \frac{1}{N_v} \left[\sqrt{[s_i(N_0 - 1) - y_i^*(N_0 - 1)]^2} - \sqrt{[s_i(N_t) - y_i^*(N_t)]^2} \right] \quad (2.113)$$

2.5.3 Diafonía (Crosstalk)

La diafonía ("Crosstalk") es un estimador que se define para dar información sobre el nivel de diferencia entre dos señales. Cuanto menor es la diafonía, mayor será la similitud entre las dos señales. Es una medida similar (inversa) a la relación señal/ruido (SNR), típicamente utilizada en procesamiento de señales; donde aquí el ruido es la diferencia entre las señales.

La diafonía de una señal $y_i^*(t)$ con respecto a $s_i(t)$ en un instante t se define como [GER94, MAN95]:

$$C_i(t) = 10 \cdot \log \left(\frac{E \left\{ (s_i(t) - y_i^*(t))^2 \right\}}{E \{ s_i^2(t) \}} \right) \quad (2.114)$$

donde el operador $E\{\cdot\}$ es la esperanza matemática.

Obsérvese que (2.114) es una medida del error (diferencia) relativo de la señal de salida con respecto a la señal de entrada. La diferencia va elevada al cuadrado para que los errores negativos no cancelen a los errores positivos.

La diafonía puede considerarse como el error cuadrático medio, pero relativo a la señal original y expresado en decibelios.

La diafonía de una señal recuperada, $y_i^*(t)$, con respecto a la fuente original, $s_i(t)$, en un instante dado, $t = N_t$, dentro de una ventana deslizante temporal $[N_0, N_t]$ se define como:

$$C_{i(v)}(N_t) \stackrel{def}{=} 10 \cdot \log \cdot \left\{ \frac{\frac{\sum_{t=N_0}^{N_t} (s_i(t) - y_i(t))^2}{N_v}}{\frac{\sum_{t=N_0}^{N_t} (s_i(t))^2}{N_v}} \right\} = 10 \cdot \log \cdot \left\{ \frac{\sum_{t=N_0}^{N_t} (s_i(t) - y_i(t))^2}{\sum_{t=N_0}^{N_t} (s_i(t))^2} \right\} \quad (dBs) \quad (2.115)$$

Denominando:

$$\begin{aligned}
 NA_i(N_f) &\stackrel{def}{=} \sum_{t=N_0}^{N_f} (s_i(t) - y_i(t))^2 \\
 DA_i(N_f) &\stackrel{def}{=} \sum_{t=N_0}^{N_f} (s_i(t))^2
 \end{aligned}
 \tag{2.116}$$

se tiene que:

$$C_{i(v)}(N_t) = 10 \cdot \log \cdot \frac{NA_i(N_t)}{DA_i(N_t)}
 \tag{2.117}$$

donde el numerador, $NA_i(N_t)$, y el denominador, $DA_i(N_t)$, del argumento se pueden obtener recursivamente con las siguientes expresiones:

$$\begin{aligned}
 NA_i(N_t) &= NA_i(N_t - 1) - [s_i(N_0 - 1) - y_i^*(N_0 - 1)]^2 + [s_i(N_t) - y_i^*(N_t)]^2 \\
 DA_i(N_t) &= DA_i(N_t - 1) - s_i^2(N_0 - 1) + s_i^2(N_t)
 \end{aligned}
 \tag{2.118}$$

Las expresiones de (2.116) a (2.118) serán utilizadas como medida e bondad de los algoritmos en los programas correspondientes, descritos en el Capítulo 3.

2.6 CONCLUSIONES

En este capítulo se han tratado de exponer sistemáticamente los fundamentos teóricos de los métodos de separación ciega de fuentes basados en propiedades algebraicas y geométricas. Esta forma de abordar el problema de separación de señales fue propuesta originalmente por investigadores del Departamento de Arquitectura y Tecnología de Computadores de la Universidad de Granada [PUN95a], y en este capítulo se han tratado de introducir de forma completa y compacta las distintas propiedades en que se fundamenta el nuevo enfoque, del que emergen distintos métodos, como los que se presentan en los Capítulos 4 y 5 de la presente memoria.

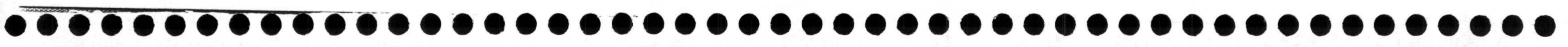
El capítulo se inicia (Sec.2.1) completando y formalizando el modelo de mezcla lineal de señales descrito inicialmente en el Capítulo 1 (Secs.1.1 y 1.2). En la Sec.2.2 se incluyen

propiedades analíticas y geométricas relacionadas con la transformación lineal de las fuentes, y que son la base de los métodos geométricos (Sec.2.3).

La idea fundamental radica en que con cualquier conjunto de p vectores de las p aristas (*vectores arista*) que incidan en un vértice cualquiera del hiperparalelepípedo contenido en el espacio de observaciones, se puede formar una matriz \mathbf{W} que estima a \mathbf{A} . La matriz \mathbf{W} tiene como elementos los vectores de las aristas ordenados en columnas.

El procedimiento descrito en el Capítulo 4 requiere estimar, además de los vectores arista, uno de los vértices. La Sec.2.4 se dedica a analizar este problema.

Por último, la Sec.2.5 incluye diversas medidas para evaluar las prestaciones: índice de Amari, error cuadrático medio entre los elementos de \mathbf{A} y \mathbf{W} , error cuadrático medio entre las fuentes originales y las estimadas, y la diafonía (“crosstalk”). Estos índices se utilizarán para medir la convergencia y calidad final de los resultados de simulación obtenidos tanto con fuentes sintéticas como reales (Capítulos 4 y 5).



CAPÍTULO 3:

ENTORNO DE SIMULACIÓN PARA DESARROLLAR ALGORITMOS DE SEPARACIÓN DE MEZCLAS LINEALES.

En el presente capítulo se presenta un conjunto de programas para la simulación de algoritmos de separación de fuentes a partir de mezclas lineales (Sec.3.1, 3.2). Los programas se han desarrollado en MATLAB, permitiendo de forma muy sencilla: (a) seleccionar las fuentes entre diversos tipos (reales o sintéticas) (Sec.3.3), (b) definir una matriz de mezcla predeterminada por el usuario o seleccionada aleatoriamente (Sec.3.4), (c) simular distintos algoritmos de estimación de la matriz de mezcla (Sec.3.5), (d) reconstruir en tiempo real las señales basándose en la utilización de una red neuronal recursiva (Sec.3.6), y (e) evaluar y visualizar las prestaciones de los resultados (Sec.3.7). El simulador permite visualizar (Sec.3.8) las diversas señales involucradas en el proceso en distintos formatos (frecuencias de amplitudes, espacios vectoriales, y en el dominio del tiempo) y entre cualquiera de las fases de simulación.

El simulador se ha utilizado para desarrollar y medir las prestaciones de los algoritmos que se describen en los Capítulos 4 y 5.

3.1. INTRODUCCIÓN

Con frecuencia se plantea la conveniencia de disponer de una herramienta software flexible y general para simular los distintos algoritmos de separación, disponiendo de fuentes tanto sintéticas como reales, y conteniendo estimadores generales para evaluar la calidad de los algoritmos. Este paquete de programas o entorno de simulación, además de tener que ser de gran modularidad para facilitar el desarrollo de nuevos algoritmos, podría ser utilizado para comparar objetivamente las prestaciones de distintos procedimientos.

En línea con lo indicado en el párrafo anterior, el presente capítulo presenta una herramienta software para la simulación de algoritmos de separación de mezclas lineales de fuentes y dispone de las características indicadas anteriormente. En efecto, el entorno de simulación, desarrollado en MATLAB, es modular y permite:

- (a) Seleccionar fuentes, entre diversos tipos (tanto reales como sintéticas),
- (b) Definir una matriz de mezcla (predeterminada por el usuario o seleccionada aleatoriamente),
- (c) Simular distintos algoritmos de estimación de la matriz de mezcla,
- (d) Reconstruir las señales originales optando por el uso de una entre tres posibilidades, basadas en la utilización de una red neuronal recursiva, y
- (e) Evaluar las prestaciones de la recuperación y reconstrucción de las fuentes, utilizando distintos estimadores generales.

Después de cada una de las cinco fases anteriores se pueden observar los resultados obtenidos, y representar gráficamente las distintas señales y estimadores involucrados en el proceso, en forma vectorial, en el dominio del tiempo, como distribución de amplitudes, o ser audio-escuchadas a través de los altavoces del PC o estación de trabajo (Sec.3.8).

Los programas simulan algoritmos de recuperación de la matriz de mezcla y de reconstrucción de las señales en tiempo real. Es decir, el valor de las señales reconstruidas en un instante t , se obtiene considerando los valores de mezcla sólo en un tiempo $\tau \leq t$, con objeto de poder ser posteriormente implementados para funcionar en tiempo real, con un hardware suficientemente rápido. El simulador se ha desarrollado haciendo especial hincapié en que la interfaz con el usuario sea muy sencilla, usando en todo momento ventanas con

menús de opciones tanto para seleccionar el algoritmo a ejecutar, como para introducir los parámetros correspondientes al mismo.

3.2. ESTRUCTURA GENERAL DEL SIMULADOR

La estructura global del simulador se puede entender fácilmente considerando las seis opciones que presenta el menú principal (Figura 3.1). La primera de ellas es para elegir el número de fuentes, p , que se desean separar, así como el número de muestras de cada señal, M (Sec.3.3). Una vez dados ambos parámetros aparece un nuevo menú con el que se pueden seleccionar, de entre distintos tipos, las fuentes, $s(t)$. Este último menú, así como las fuentes seleccionables, se describen en la Sec.3.3.

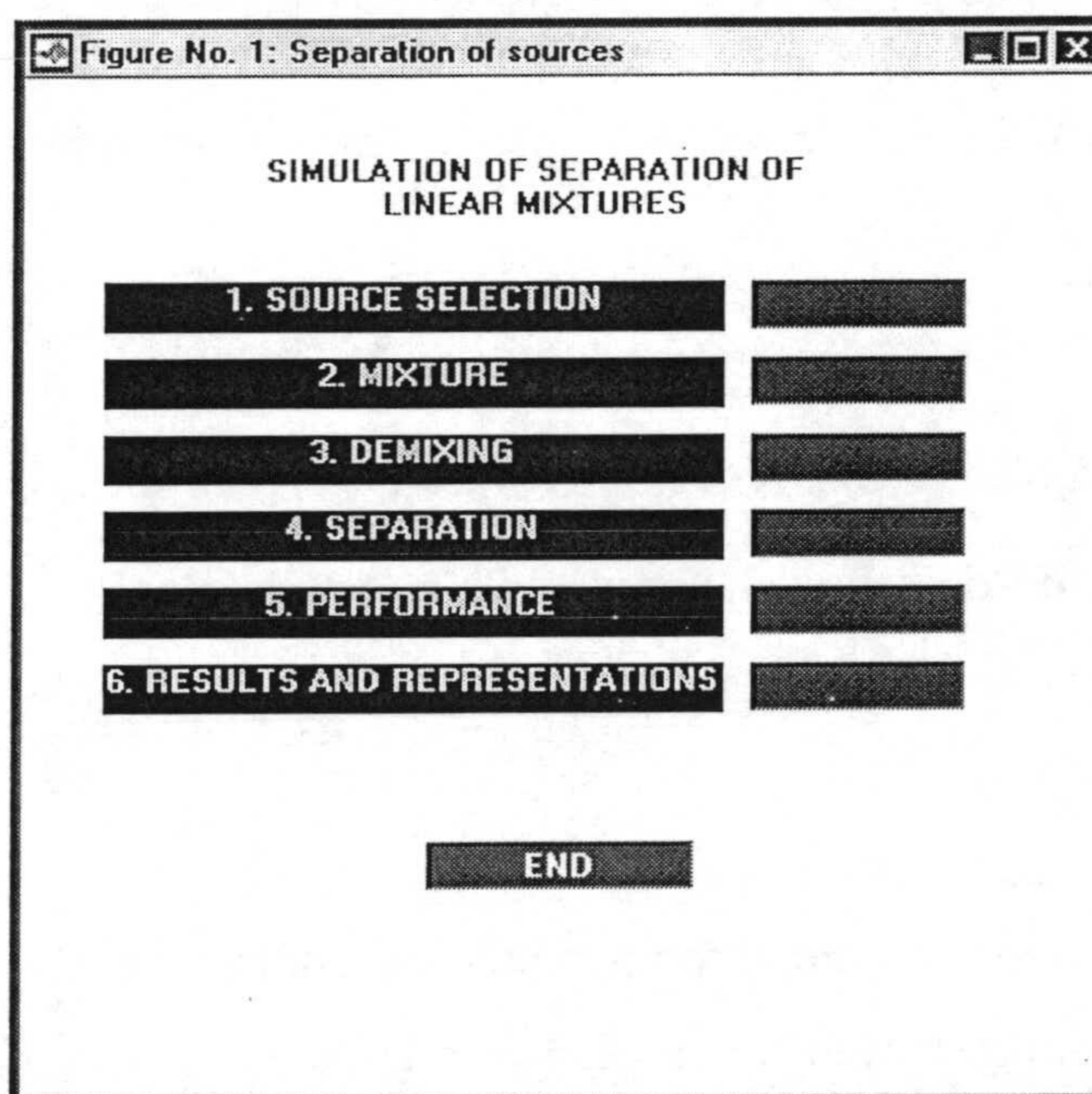


Figura 3.1. Menú principal del simulador.

La segunda opción despliega un menú para establecer la matriz de mezcla A (Sec.3.4), que se puede introducir elemento a elemento o elegir aleatoriamente, y, al pulsar el botón de *aceptar*, se lleva a cabo la mezcla lineal (1.11), (2.2), de las fuentes seleccionadas previamente, obteniéndose las señales $e(t)$.

La tercera opción permite elegir varios métodos para estimar la matriz \mathbf{A} ; es decir, permite obtener una matriz \mathbf{W} , relacionada con \mathbf{A} , según (1.12) y (2.10), conociendo únicamente las mezclas, $e(t)$ (Sec.3.5).

La cuarta opción sirve para seleccionar un algoritmo de reconstrucción de las fuentes (Sec.3.6). La selección puede efectuarse de acuerdo con la naturaleza de las fuentes (fuentes que varían lentamente con el tiempo, o no).

Para poder valorar los resultados obtenidos, la opción número 5 del menú principal permite seleccionar distintos estimadores que miden las prestaciones de los resultados (Sec.3.7).

Finalmente, la última opción del menú principal (Sec.3.8) ofrece la posibilidad de visualizar los resultados de cada fase, mediante representaciones gráficas de las señales (originales, mezclas y recuperadas), de los índices de prestaciones, definidos en la Sec.2.5, y otros resultados relevantes.

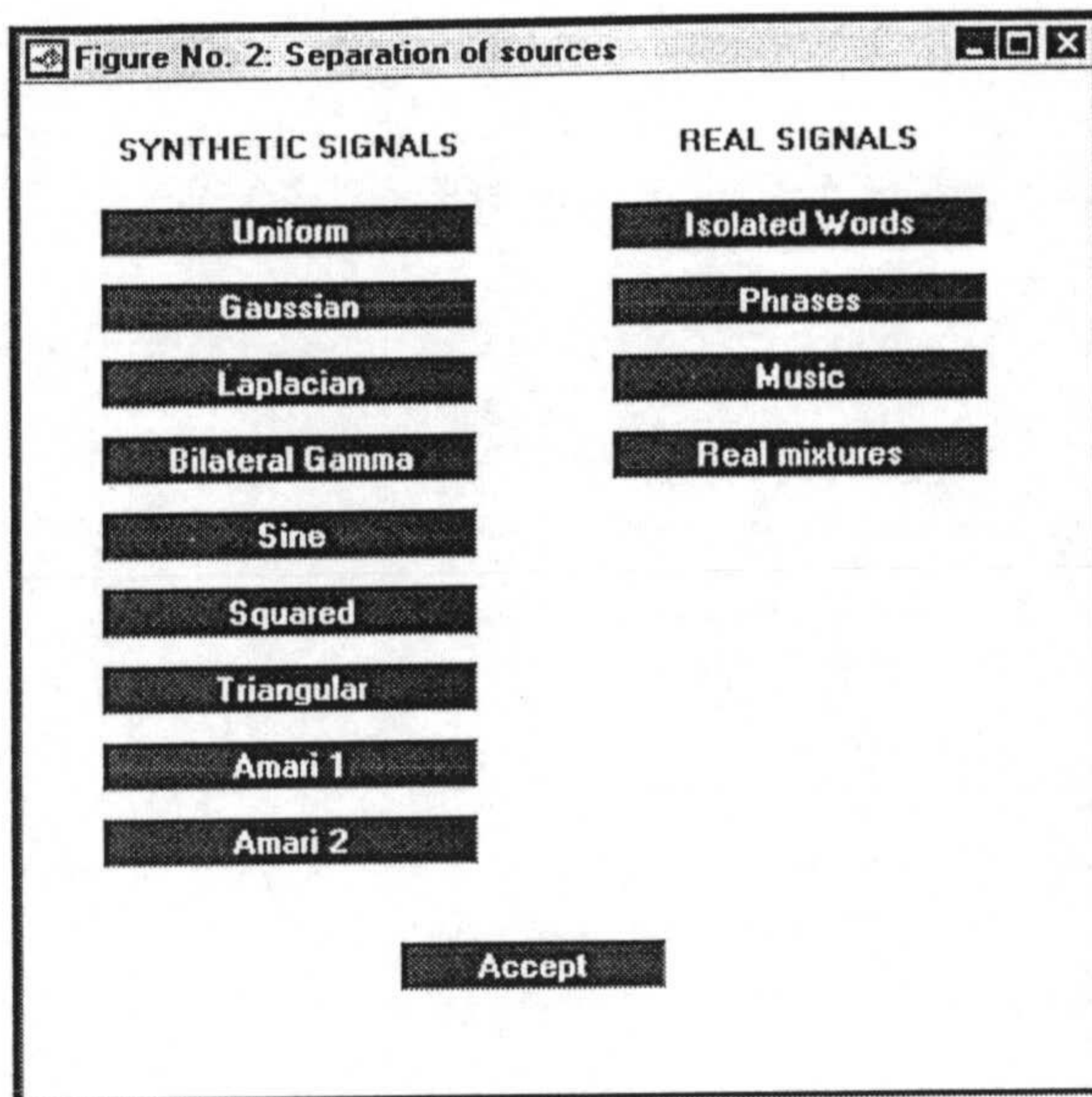
3.3. MÓDULO DE GENERACIÓN DE SEÑALES

En el menú de generación de señales (Figura 3.2) se puede elegir entre dos tipos de señales: sintéticas y reales. Antes de elegir las fuentes el usuario tiene que dar el número de fuentes y de muestras con las que se desea llevar a cabo la mezcla (Figura 3.2a).

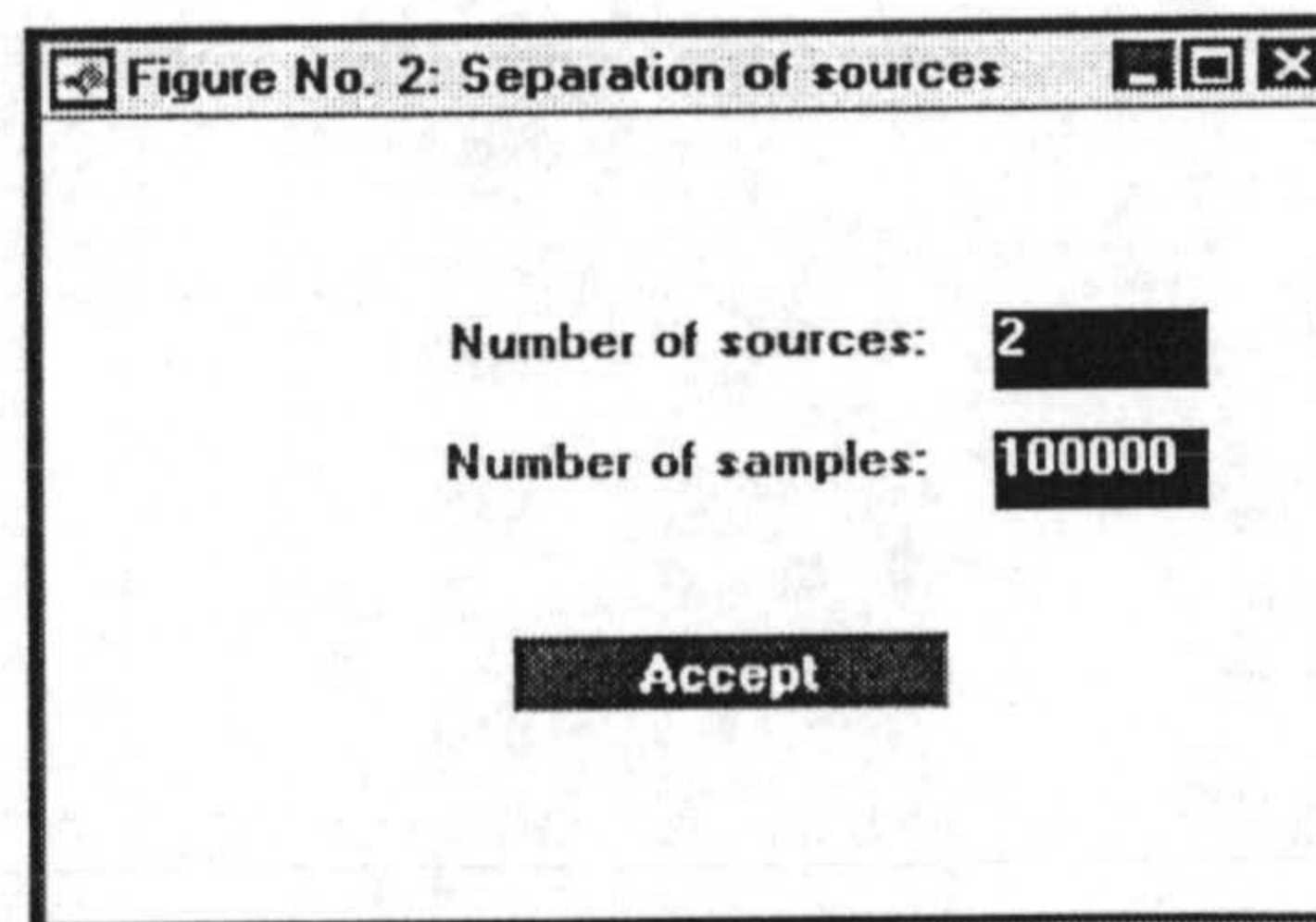
A continuación se describen brevemente las distintas señales que puede generar el simulador como fuentes, en su configuración actual. En primer lugar (Sec.3.3.1) se tratan las señales sintéticas, y posteriormente (Sec.3.3.2) se incluyen las señales reales.

3.3.1 Señales sintéticas disponibles en el simulador

Dentro de las señales sintéticas se encuentran las siguientes: aleatoria con distribución uniforme, gaussiana, laplaciana, gamma bilateral, seno, cuadrada, triangular, y dos señales que se han denominado Amari1 y Amari2. A continuación se indica como se genera cada una de estas señales.



(b)



(a)

Figura 3.2. Menú de generación de señales.

Generación de señales gaussianas

En la Figura 3.3 se muestra una función de una distribución normal o gaussiana. Para más detalles puede consultarse la Sec.A1.3.2 del Apéndice 1.

A partir de un conjunto de señales aleatorias, $\mathbf{u}=(u_i)$, uniformemente distribuidas se puede generar fácilmente una señal aleatoria gaussiana, *gaussi*, por ejemplo, utilizando el promedio de ellas [BUR98], o bien evaluando la siguiente expresión [PRO97]:

$$C = \sqrt{2 \cdot \sigma^2 \cdot \ln[1/(1 - u_1)]} \cdot \cos(\theta) \quad (3.1)$$

donde u_1 es una variable aleatoria uniformemente distribuida en el intervalo [0,1], θ es otra variable uniformemente distribuida en el intervalo [0,2 π] y σ es la varianza deseada para la función gaussiana a obtener.

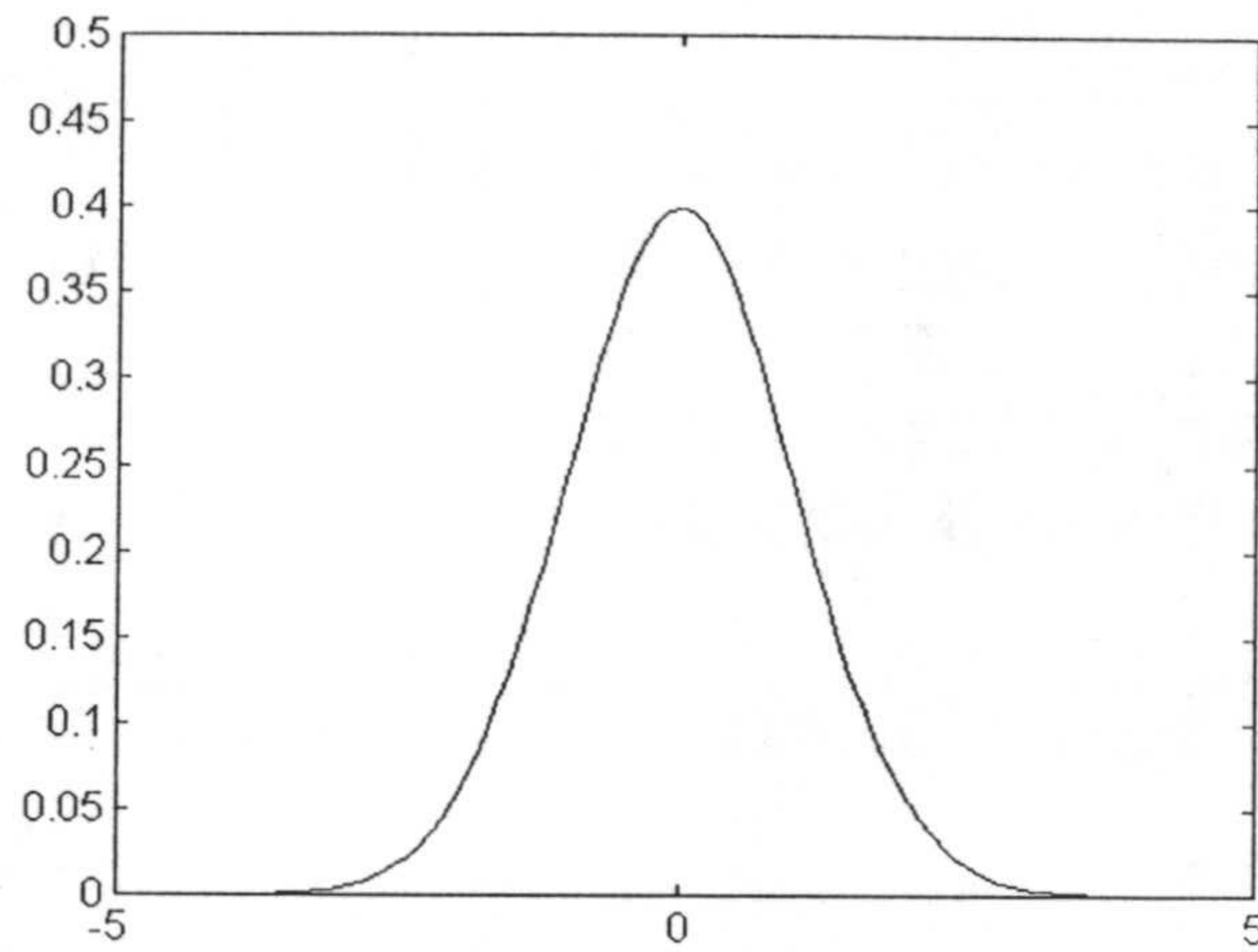


Figura 3.3. *Función de distribución Gaussiana.*

No obstante, como el entorno de desarrollo del simulador (MATLAB) dispone de la función *randn* que genera valores aleatorios con una distribución gaussiana estandarizada, es decir, con $\sigma=1$ y media cero ($\mu=offset=0$), es más fácil utilizar el siguiente código:

```
gausi=(ds*ones(1,M))*randn(1,M)+offset;
```

donde M es el número muestras, *ds* y *offset* son respectivamente la desviación estándar y el valor medio de la señal aleatoria a generar. Estos últimos parámetros son solicitados por el programa cuando el usuario selecciona como fuente una señal gaussiana (Figura 3.4a).

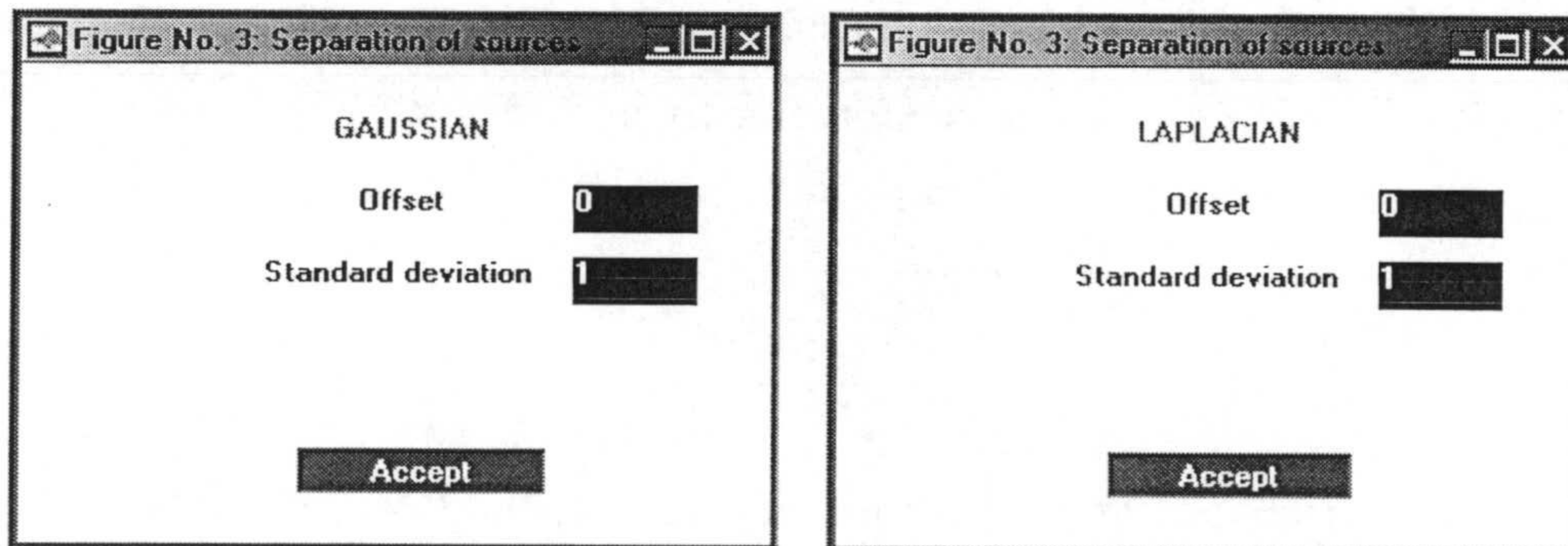


Figura 3.4. *Parámetros de las señales Gaussiana y Laplaciana.*

Señal aleatoria con función de distribución de probabilidad laplaciana.

Cuando se solicita al simulador la producción de una señal aleatoria laplaciana (Sec.A1.3.4, Figura 3.5), es decir, un conjunto de M muestras con la siguiente función de distribución:

$$f(s) = \frac{1}{\sqrt{2} \cdot \sigma} \cdot e^{-\sqrt{2} \cdot |s-\mu| / \sigma} \quad (3.2)$$

se obtienen evaluando la siguiente expresión [JOH94], [BUR97]:

$$s = \begin{cases} \frac{1}{\alpha} \log(2u_1) & \text{si } 0 < u_1 \leq \frac{1}{2} \\ -\frac{1}{\alpha} \log\{2(1-u_1)\} & \text{si } \frac{1}{2} < u_1 < 1 \end{cases} \quad (3.3)$$

donde $\alpha = \sqrt{2} \cdot \sigma$, y u_1 son números aleatorios uniformemente distribuidos en el intervalo $[0,1]$.

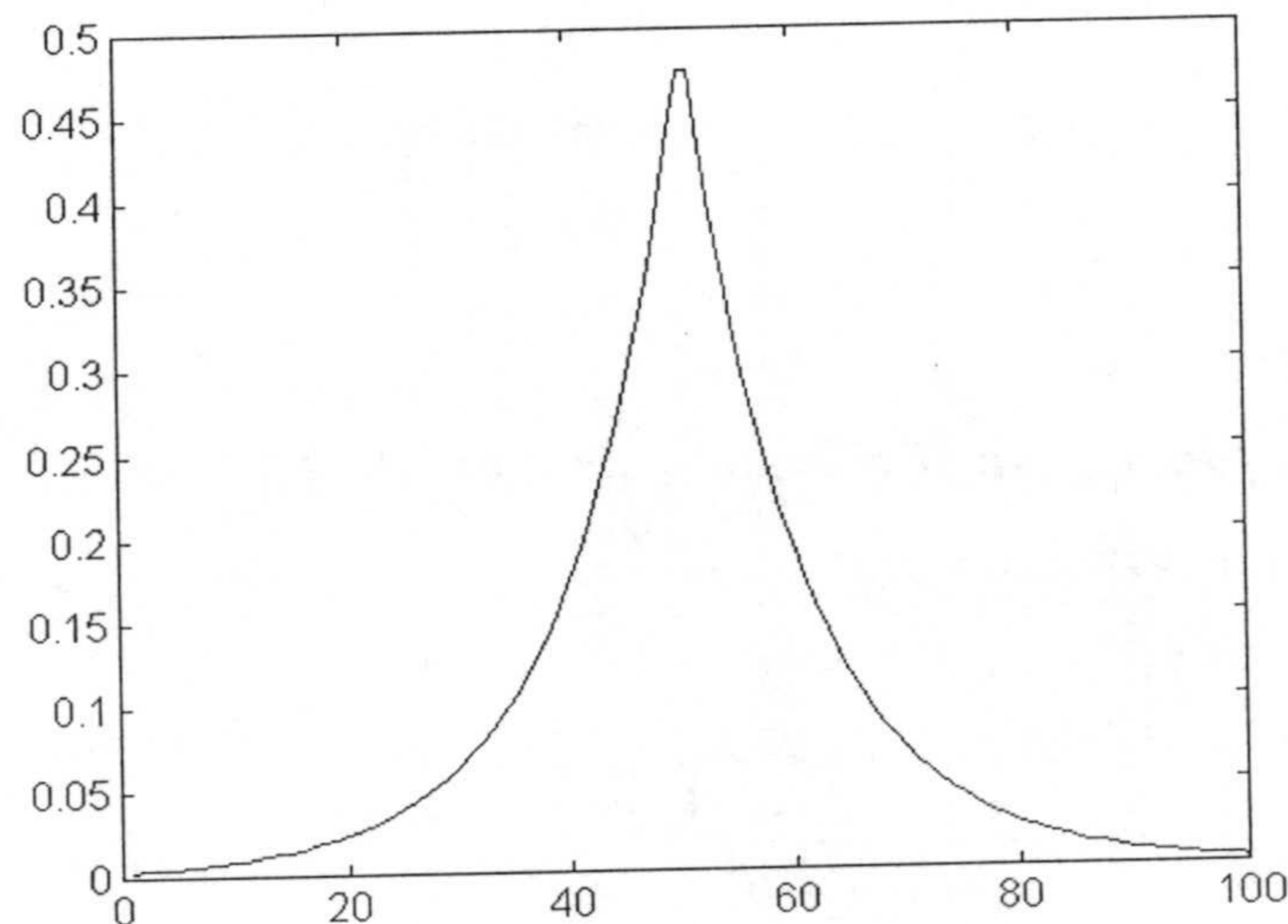


Figura 3.5. Función de distribución Laplace ($\mu=0$, $\sigma=\sqrt{2}$).

Al igual que para la señal gaussiana, para generar una laplaciana se solicita el valor medio (“offset”) y la desviación estándar (Figura 3.4b). El código en MATLAB para generar dicha señal es:

```
lap=offset+(x*dst);
```

donde *dst* y *offset* son respectivamente la desviación estándar y el valor medio; *x* se genera a su vez según (3.3).

En la Figura 3.6b se muestran las funciones de distribución de dos señales laplacianas y gaussianas generadas por el simulador.

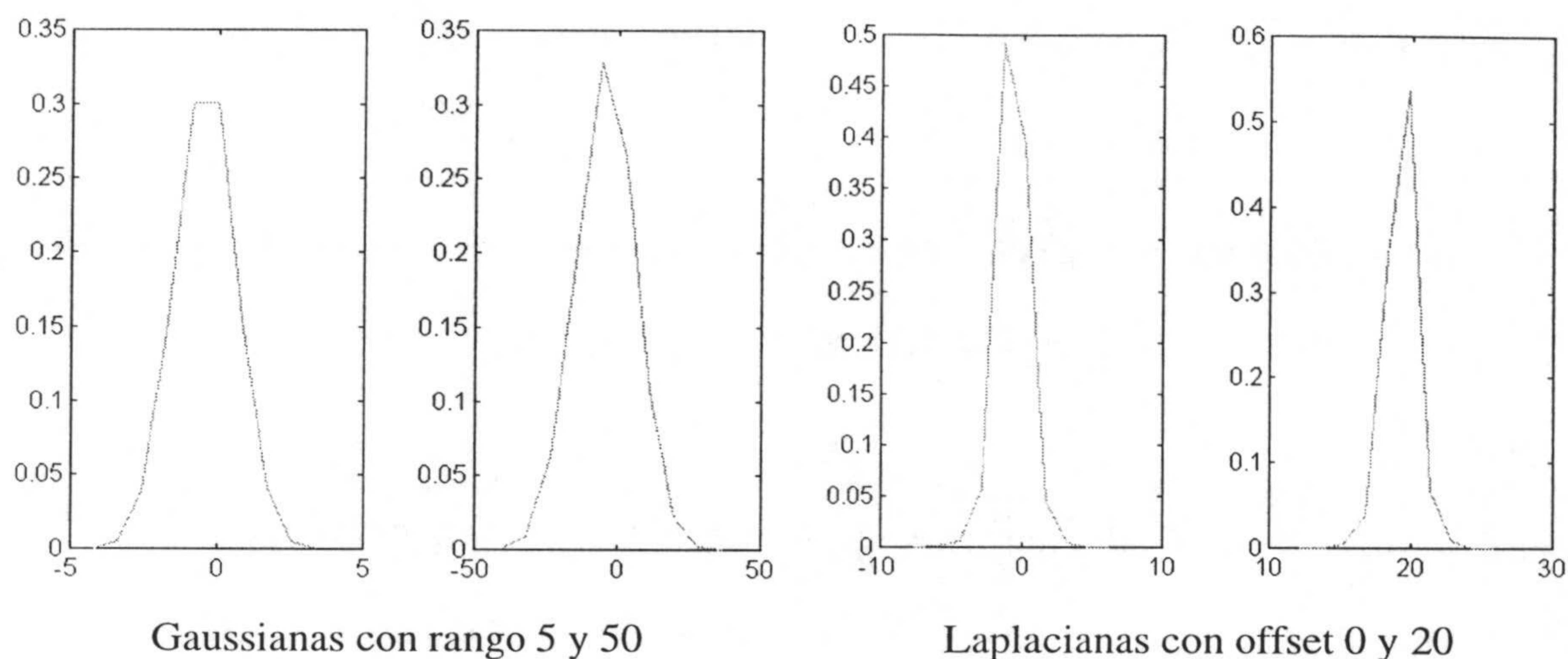


Figura 3.6. Ejemplos de señales: (a) Gaussianas con rango 5 y 50 (b) Laplacianas con offset 0 y 20, y.

Señal aleatoria con función de distribución gamma bilateral

Una señal aleatoria cuya función de distribución es:

$$f(s) = \left(\frac{\sqrt{3}}{8\pi\sigma|s-\mu|} \right)^{1/2} \cdot e^{-\sqrt{3}|s-\mu|/2\sigma} \quad -\infty \leq s \leq \infty \quad (3.4)$$

se dice que es de tipo gamma bilateral (Sec.A1.3.3).

A diferencia de las señales aleatorias gaussiana y laplaciana no hay forma directa sencilla de obtener una señal aleatoria cuya función de distribución sea de tipo gamma [JOH94], por lo que aquí se ha optado por seguir los pasos que se indican a continuación:

1. *Generación de muestras ordenadas por su valor y que siguen una distribución de probabilidad gamma bilateral.*

Suponiendo que se desea generar N muestras de una señal con función de probabilidad (3.4) y estandarizada (es decir, $\mu=0$, $\sigma=1$, Sec.A1.2.2) se puede obtener de (3.4) el

número de muestras, $n(\tilde{s})$, que debe haber de cada valor estandarizado, \tilde{s} , de la siguiente forma:

$$n(\tilde{s}) = N \frac{0.2625}{\sqrt{\tilde{s}}} e^{-0.8660|\tilde{s}|} \quad (3.5)$$

La expresión anterior (3.5) indica que si se desea obtener un conjunto de N muestras estandarizadas que sigan una distribución gamma bilateral (3.5), debe haber $n(-9)$ muestras del valor $\tilde{s} = -9$, $n(-8)$ muestras del valor $\tilde{s} = -8$, etc.

Dando valores a (3.5) y haciendo $N=400000$, se observa que $n(\tilde{s}) \geq 5$ sólo dentro del intervalo $-8 < \tilde{s} < 8$.

Teniendo en cuenta los conceptos anteriores se ha desarrollado un programa en MATLAB (“gam_ma4.m”) que aplica la expresión (3.5) a los valores $\tilde{s} = -8, -7.99, -7.98, \dots, 0, \dots, 7.98, 7.99, 8$ y genera un fichero (“mad_gam2”) en el que se incluye $n(-8)$ veces el valor $\tilde{s} = -8$, $n(-7.99)$ veces el valor $\tilde{s} = -7.99$, y así sucesivamente. De esta forma, por construcción, se obtienen N muestras cuya distribución de probabilidad es (3.4). Ahora bien, las muestras no son aleatorias en el tiempo, ya que sus valores están ordenados de menor a mayor, $\tilde{s} = -8, -7.99, -7.98, \dots, 0, \dots, 7.98, 7.99, 8$.

2. Aleatorización del orden de las muestras.

Es fácil desarrollar aleatoriamente (barajar) las posiciones en el tiempo de los valores de las muestras. En MATLAB, por ejemplo, se realiza con el siguiente código que utiliza la función *randperm*:

$$s = s_ord(randperm(400000))$$

Donde s es el vector que contiene la señal aleatoria estandarizada deseada y s_ord es el vector con las muestras ordenadas.

3. *Generación de la señal solicitada.*

Cuando el usuario selecciona como una de las fuentes una señal aleatoria gamma bilateral, el simulador a través de una ventana solicita la media (μ) y la desviación estándar (σ) que se desean, y lleva a cabo las siguientes acciones:

- (a) Lee el fichero "mad_gam2"
- (b) Baraja los elementos del fichero (según se indicó en el paso 2)
- (c) Selecciona las $M \leq N$ primeros elementos del fichero, donde M es el número de muestras solicitadas. Así se tienen las M muestras estandarizadas, \tilde{s} .
- (d) Efectúa la siguiente transformación (A1.1.4):

$$s = \sigma \cdot \tilde{s} + \mu$$

Con este procedimiento se obtiene la señal deseada.

Generación de señales periódicas.

El simulador puede generar las señales periódicas que se indican a continuación.

- *Señal seno:* el simulador pide al usuario el periodo de la señal. El código en MATLAB es el siguiente:

```
x=(0:T:(M+1)*T);
```

```
Seno=5*sin(x);
```

donde T es el periodo de la señal seno y M el número de muestras.

- *Señal cuadrada:* el simulador pide al usuario el periodo de la señal. El código en MATLAB es el siguiente:

```
vm=10; tt=0;
```

```
for t=1:M
```

```
    sc(l,t)=vm;
```

```
    tt=tt+1;
```

```
    if (tt > T/2)
```

```
        tt=0;
```

```
        if vm==10
```



```

                                vm=0;
                                else
                                vm=10;
                                end
                                end
                                end
                                end
                                end

```

donde T es el periodo de la señal seno y M el número de muestras.

- *Señal triangular*: el simulador también pide al usuario el periodo de la señal. El código en MATLAB es el siguiente:

```

tri=zeros(1,M);
to=0;
k=20/T;
for t=1:M
    tri(1,t)=k*(t-to);
    if tri(1,t) >= 10
        k=-k;
        to=t+1;
    end
    if tri(1,t)<=0
        k=-k;
        to=t;
    end
end
end
end

```

Señales especiales.

También se ha considerado de interés incluir dos funciones propuestas por Amari y Cichocki [AMA95], y que son utilizadas por diversos autores:

$$Am_1(t) = 5 \cos\left(\frac{2\pi}{T_1}\right) \text{sen}\left(\frac{2\pi}{T_2}\right) \quad (3.6)$$

$$Am_2(t) = 0.01 \left[\text{sign} \left(\sin \frac{2\pi}{T_1} t \right) + 9 \cos \left(\frac{2\pi}{T_2} t \right) \right] \quad (3.7)$$

que en el menú aparecen con las denominaciones Amari1 y Amari2, respectivamente y sus códigos en MATLAB son:

- *Amari1* (Figura 3.7a):

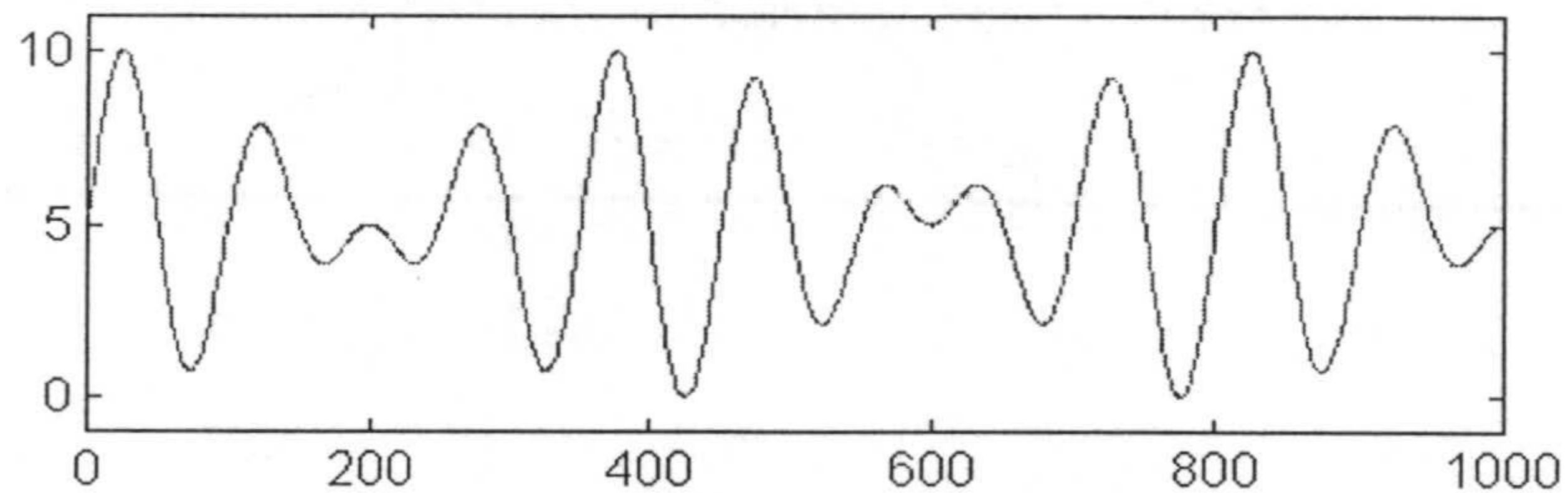
```
omega1=2*pi/800;
omega2=2*pi/100;

for t=1:M
    am1(1,t)=5*cos(omega1*t)*sin(omega2*t);
end
```

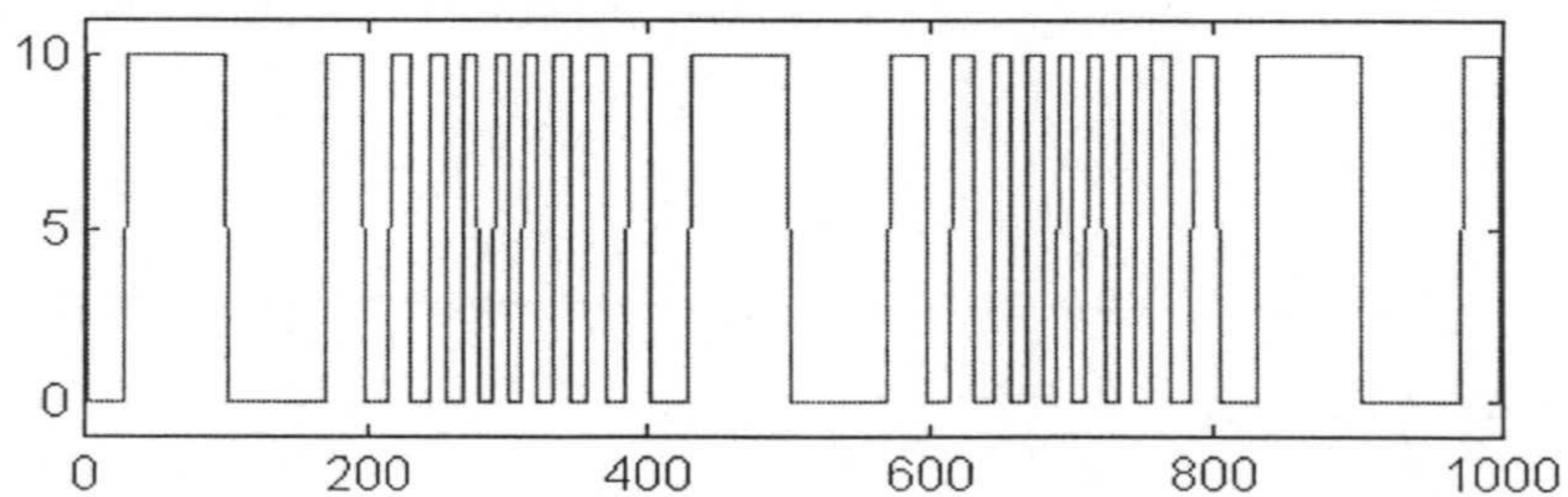
- *Amari2* (Figura 3.7b):

```
alfa1=2*pi/400;
alfa2=2*pi/40;

for t=1:M
    am2(1,t)=0.01*(sign(sin(alfa1*t+9*cos(alfa2*t))));
end
```



(a)



(b)

Figura 3.7. Ejemplos de señales (a) *Amari1*, y (b) *Amari2*.

3.3.2 Señales reales incluidas en el simulador.

Dentro de las señales reales se puede elegir entre tres tipos: palabras aisladas, frases procedentes de distintas personas y fragmentos de música.

En la Figura 3. 8.b se puede ver la lista de palabras aisladas que se pueden escoger. Estas señales corresponden a un corpus creado para un sistema de control por voz para un robot de brazo articulado [PEI91]. Las muestras fueron capturadas a una frecuencia $F_s=8$ KHz con un conversor A/D de 14 bits, y tienen una relación señal/ruido de 24 dBs. Las palabras “mano” y “ocho” están compuestas por 2250 muestras (0.28125 segundos) y las palabra “dedos” de 2500 muestras (0.31250 segundos).

En la Figura 3. 8.a aparecen los nombres de las personas que han leído una frase que ha sido grabada a través de un micrófono. Tres de ellas (Juliette, Orson Wells y Pulp Fiction) se han tomado de CDs.

En la Figura 3. 8.c aparece también la referencia a dos pequeños fragmentos de Beethoven (el inicio de la 5ª sinfonía) y Mozart (un fragmento del Réquiem) tomadas también de CDs.

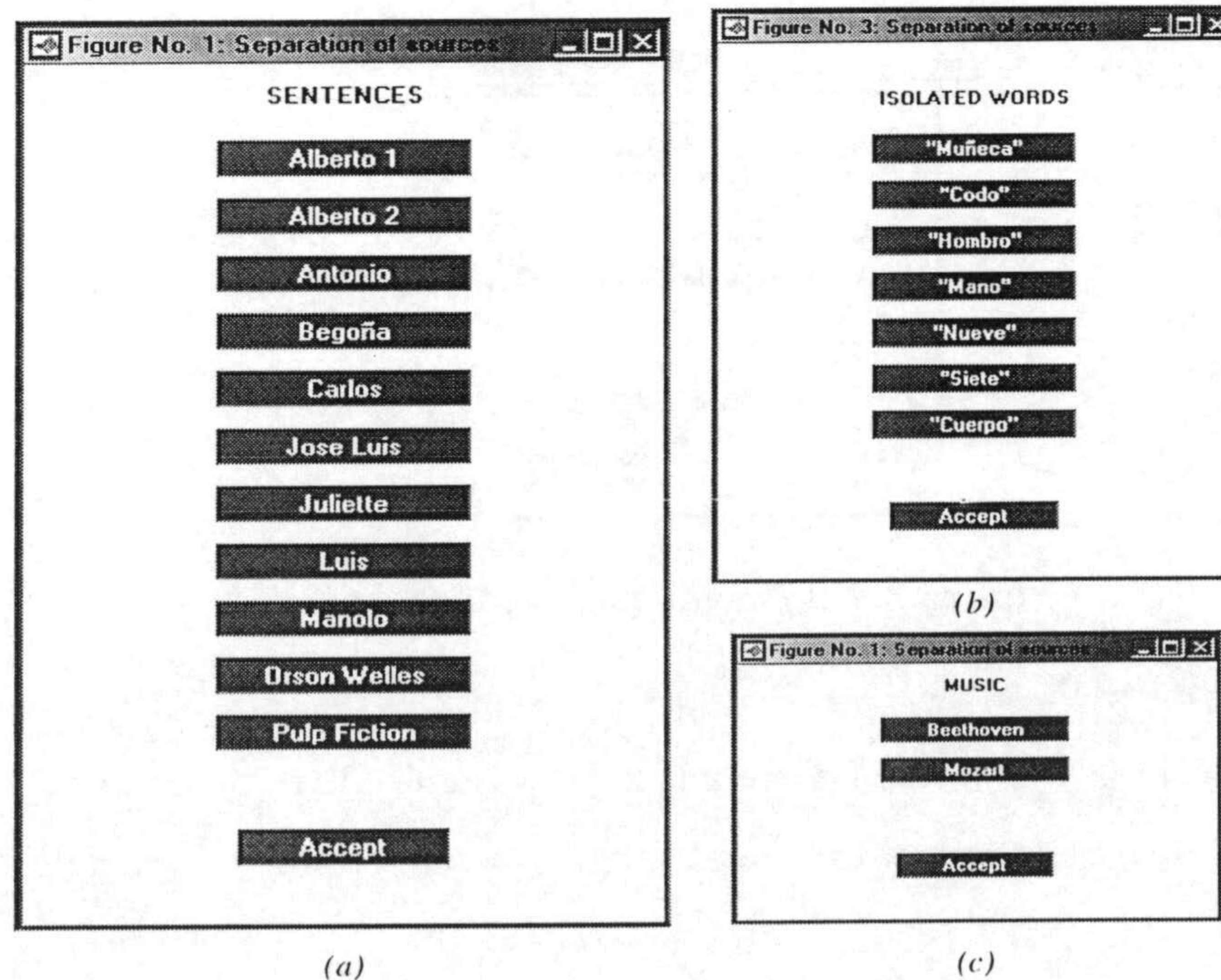


Figura 3. 8. Menú con las señales reales.

En el Capítulo 5 (Sec.5.4.2, Tabla 5.7) se describen las características de las frases y fragmentos musicales.

3.4. MÓDULO PARA SIMULAR LA MEZCLA

Como se dijo en la Sec.3.2, este módulo posibilita elegir la matriz A , hacer la mezcla según (1.11) y poder añadir ruido aditivo (3.6) que modele la corrupción de las señales mezcladas con ruido interno al canal; es decir, el ruido no se considera aquí como una fuente más.

En la ventana correspondiente a este módulo (Figura 3.9) aparecen unos valores de la matriz de mezcla por defecto. El usuario puede cambiar a su gusto dichos valores introduciéndolos manualmente, o también existe la posibilidad de elegir valores aleatorios (entre -1 y 1) para dicha matriz pulsando el botón "random A". Al pulsar el botón de "Accept" se realiza la mezcla automáticamente con los valores seleccionados.

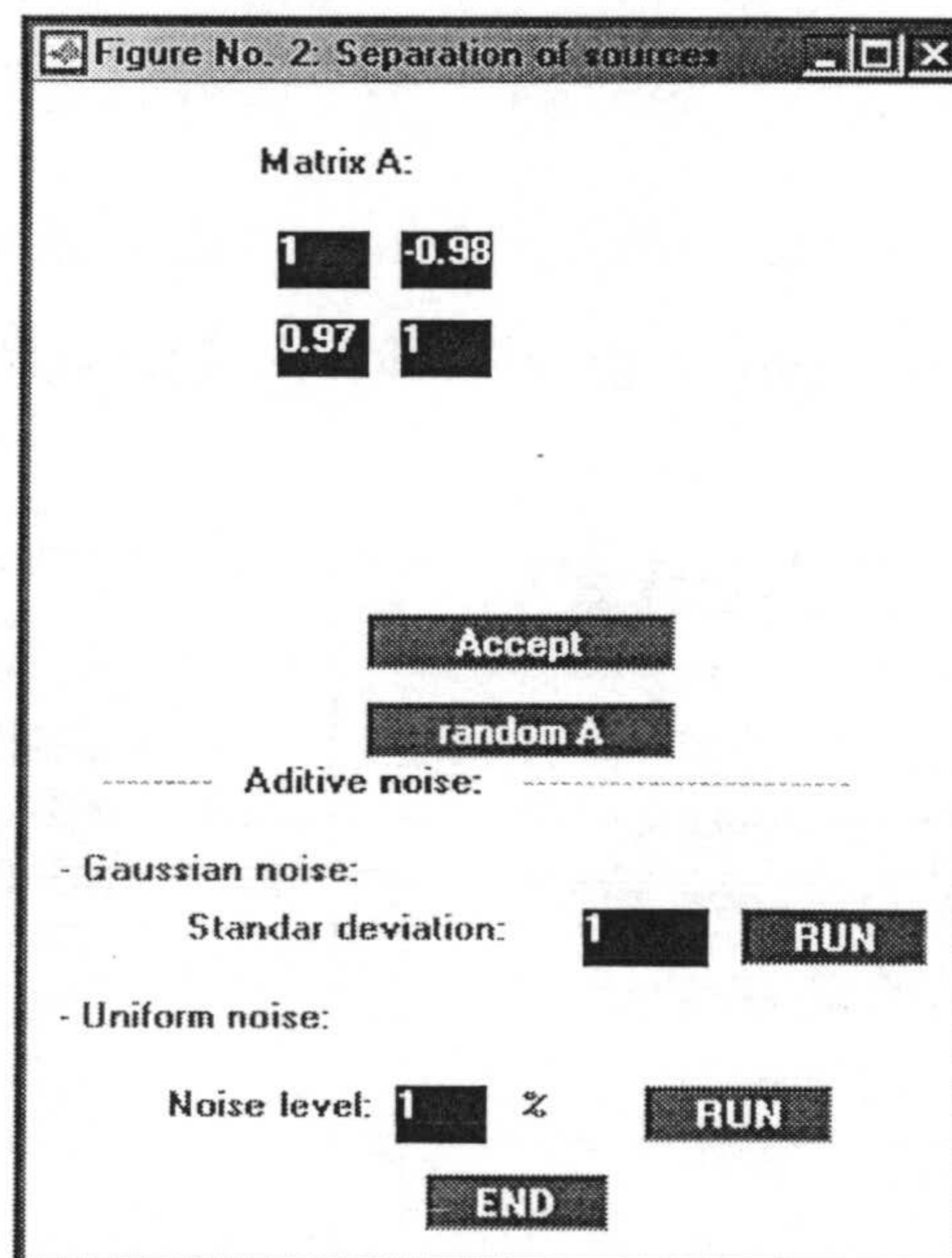


Figura 3.9. Elección de la matriz de mezcla A .

La parte inferior de la ventana se refiere a la superposición de ruido en las señales mezcladas, de acuerdo con el siguiente modelo:

$$e(t) = As(t) + n(t) \quad (3.8)$$

donde $\mathbf{n}(t)=(n_i(t))^T$, $i=1,\dots,p$, es un vector cuyas componentes representan el ruido añadido a cada una de las señales. El tipo de ruido puede seguir una distribución o bien gaussiana, en cuyo caso hay que especificar la desviación estándar o bien uniforme, dando el nivel de ruido que se desea introducir. El ruido se añade una vez se pulsa el botón “RUN”.

Tanto si se desea añadir ruido como si no, hay que pulsar el botón “END” para que se realice la mezcla y se añada o no ruido.

3.5. MÓDULO PARA LA ESTIMACIÓN DE LA MATRIZ DE MEZCLA

Este módulo (Figura 3.10) ofrece tres posibilidades para estimar la matriz de mezcla, W . Se pueden usar la red neuronal competitiva descrita en el Capítulo 4 (“competitive” en la Figura 3.10), el método de distribución angular referido en el Capítulo 5 (“Alpes” en la Figura 3.10), y una variación de este último para el caso de mezclas reales (“Real mixtures” en Figura 3.10).

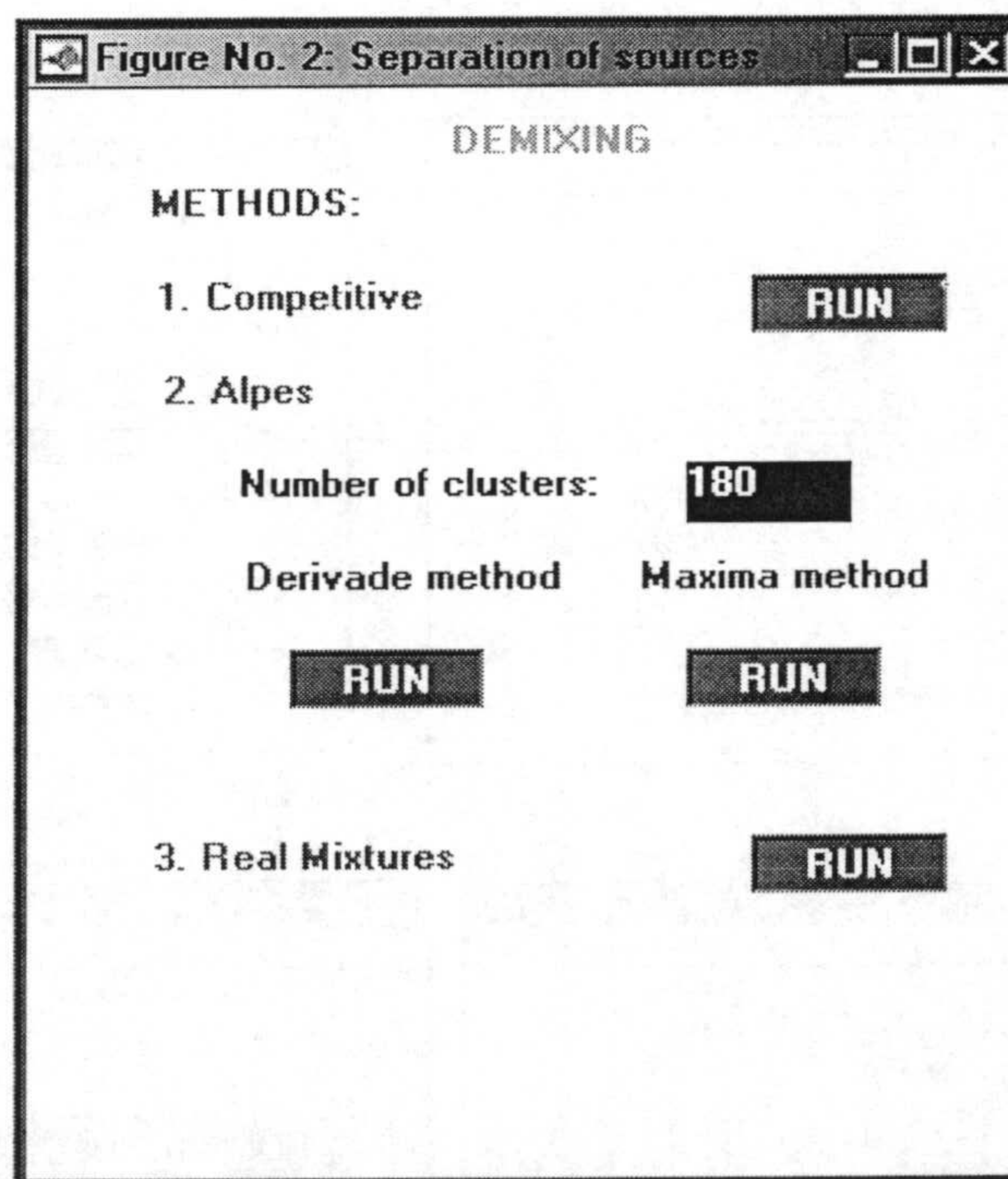


Figura 3.10. Formas de recuperación de la matriz de mezcla.

El método de distribución angular se basa en la consideración de un número predeterminado de clases (*clusters*), n_c , cada uno de ellos representado por una neurona. El programa permite seleccionar arbitrariamente el número de clases a considerar (“number of clusters”, en Figura

3.10). Dentro de este método se tienen a su vez dos opciones para obtener los máximos de la distribución (Sec.5.3.2), uno mediante la derivada o el otro mediante máximos. Una vez elegido el método, desaparece la ventana actual y surgen dos nuevas en las que se pueden visualizar el espacio de vectores de mezcla junto con los vectores que representan las columnas de la matriz W en una (Figura 3.11a), y en la otra el tiempo que va transcurriendo (en segundos), el valor de las matrices W y A (en forma canónica, Sec.2.5.2.1), y el error absoluto entre los elementos de ambas matrices (Figura 3.11b, Sec.2.5.2.1). De esta forma se puede ir comprobando visualmente la evolución del proceso de estimación de la matriz de mezcla. Una vez que termina el proceso de estimación de la matriz A al pulsar cualquier tecla aparece la distribución angular destacando los dos máximos de la distribución (Figura 3.11c). En el caso de escoger el método de la derivada aparecerá otro gráfico con la derivada de la distribución angular (Sec.5.3.2, Figura 3.11).

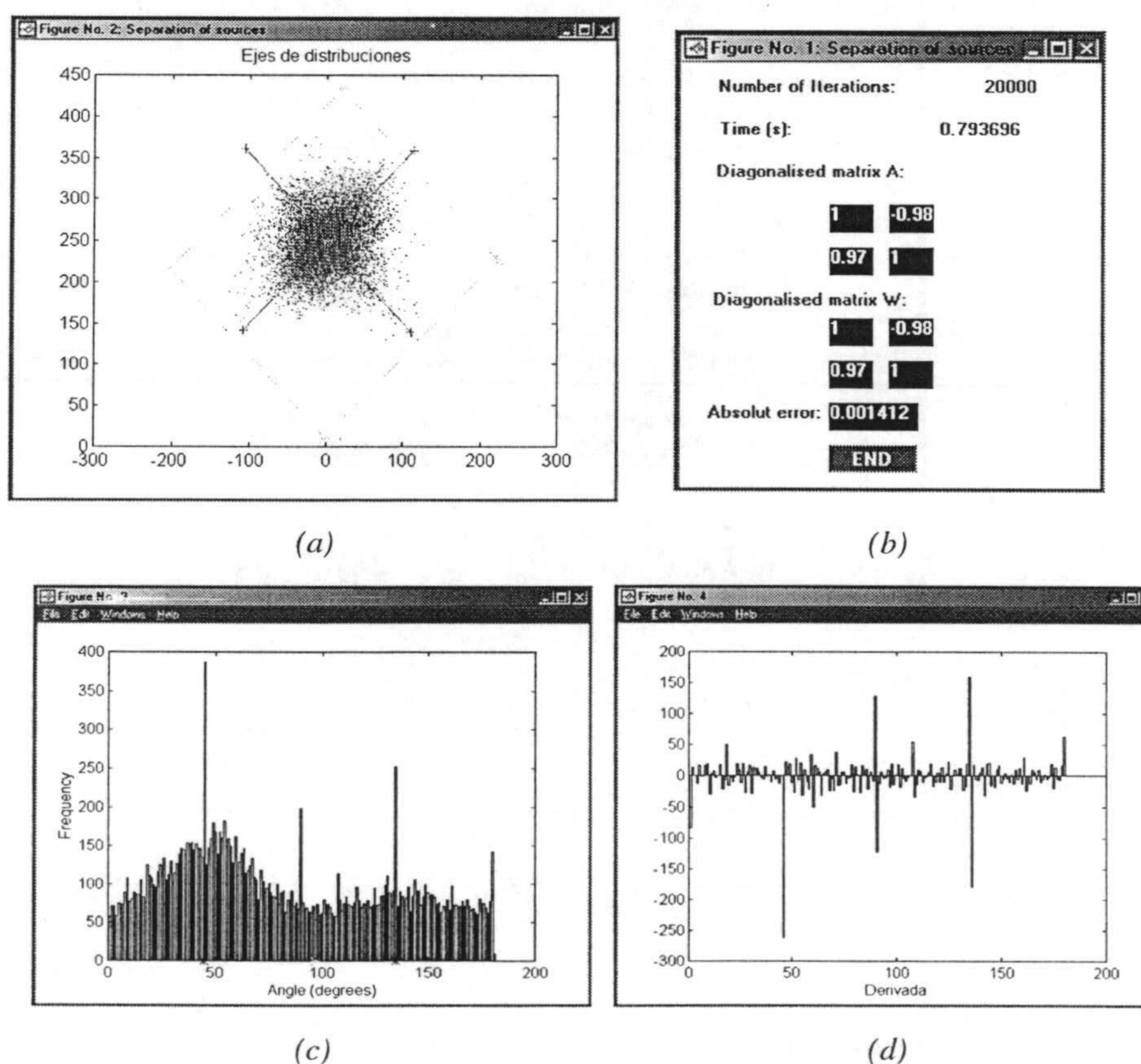


Figura 3.11. Evolución del proceso de estimación de la matriz A . (a) Espacio de mezclas, (b) Ventana de resultados, (c) Máximos de la distribución, (d) Derivada.

3.6. RECONSTRUCCIÓN DE LAS FUENTES

El simulador en su configuración actual permite reconstruir las fuentes originales (es decir, obtener $y(t)$, (2.8), según cuatro métodos (Figura 3.12). El primero de ellos, el más general, utiliza una red neuronal recursiva (Sec.1.4, 1.6 y 1.12.1) para reconstruir las señales, utilizando como entradas los valores de las mezclas, $e(t)$. Denominando m al número de muestra, es decir $t=mT_s$, las salidas obtenidas en la red neuronal son:

$$y_i(m) = \frac{1}{w_{ii}(m)} \cdot e_i(m) - \sum_{j=1, j \neq i}^p \frac{w_{ij}(m)}{w_{ii}(m)} y_j(m) \quad \forall i \in \{1, \dots, p\} \quad (3.9)$$

que corresponden a los valores de y_i despejados en (2.2). La ecuación anterior (3.9) debe aplicarse iterativamente para cada valor de $t=mT_s$, ya que $y_i(m)$ es función de los otros valores de y : $y_j(m)$, $j \neq i$. Se parte de un valor inicial, $y_i(m)=e_i(m)$, y se obtiene un nuevo valor de $y_i(m)$ aplicando (3.9). Este nuevo valor se sustituye en el segundo miembro de (3.9) y al cabo de un número determinado de iteraciones la diferencia entre dos valores consecutivos de $y_i(m)$ es menor que una cota de error determinada, considerándose que el proceso iterativo debe concluir por haberse obtenido un valor de $y_i(m)$ con suficiente precisión. El menú solicita el número de iteraciones a efectuar con cada muestra que se considera adecuado para la precisión que se desee ("iterations per sample"). Por defecto se toma el valor 14, con el que se ha obtenido siempre una convergencia razonable (con señales de voz se obtiene un error entre iteraciones sucesivas de 0.0001, Sec.4.7, Figura 4.17).

El segundo método se puede aplicar sólo a señales que varían lentamente con el tiempo. En este caso se supone que:

$$y(t) \approx y(t - \tau) \quad (3.10)$$

o discretizando el tiempo:

$$y(m) \approx y(m - 1) \quad (3.11)$$

con lo que en el segundo miembro de (3.9), $y_j(m)$, se puede sustituir por $y_j(m-1)$:

$$y_i(m) = \frac{1}{w_{ii}(m)} e_i(m) - \sum_{j=1, j \neq i}^p \frac{w_{ij}(m)}{w_{ii}(m)} y_j(m-1) \quad \forall i \in \{1, \dots, p\} \quad (3.12)$$

De esta forma el proceso de reconstrucción es mucho más rápido ya que no es necesario realizar iteraciones para cada instante $t=mT_s$, con objeto de obtener un valor correcto de $y_i(m)$.

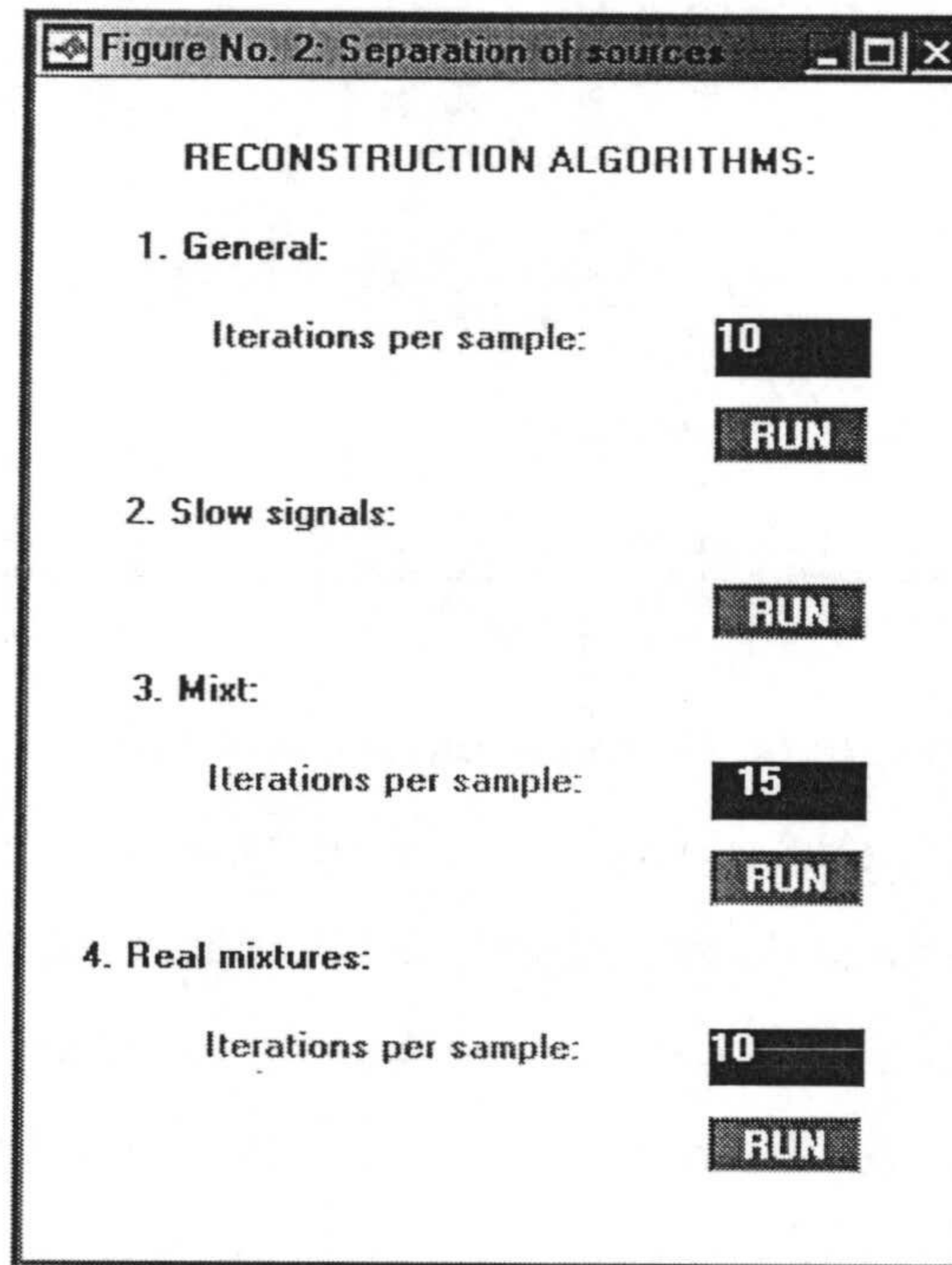


Figura 3.12. Algoritmos de reconstrucción de las fuentes.

El tercer procedimiento combina, de forma satisfactoria los dos anteriores: para cada valor de $t=mT_s$ se obtienen unos valores iniciales de $y_i(m)$ aplicando la expresión (3.12) y dichos valores se refinan aplicando iterativamente la ecuación (3.9). En este caso, el menú solicita al usuario el número de iteraciones a realizar por muestra.

El último método, para mezclas reales, es una adaptación del primer método, que actualmente estamos desarrollando.

3.7. MÓDULO DE MEDIDAS DE LAS PRESTACIONES

Otro de los módulos (Figura 3.13) de que consta el simulador permite calcular distintos estimadores con el fin de tener una medida objetiva de los resultados de los procesos de obtención de una matriz \mathbf{W} similar a \mathbf{A} y de reconstrucción de las fuentes originales.

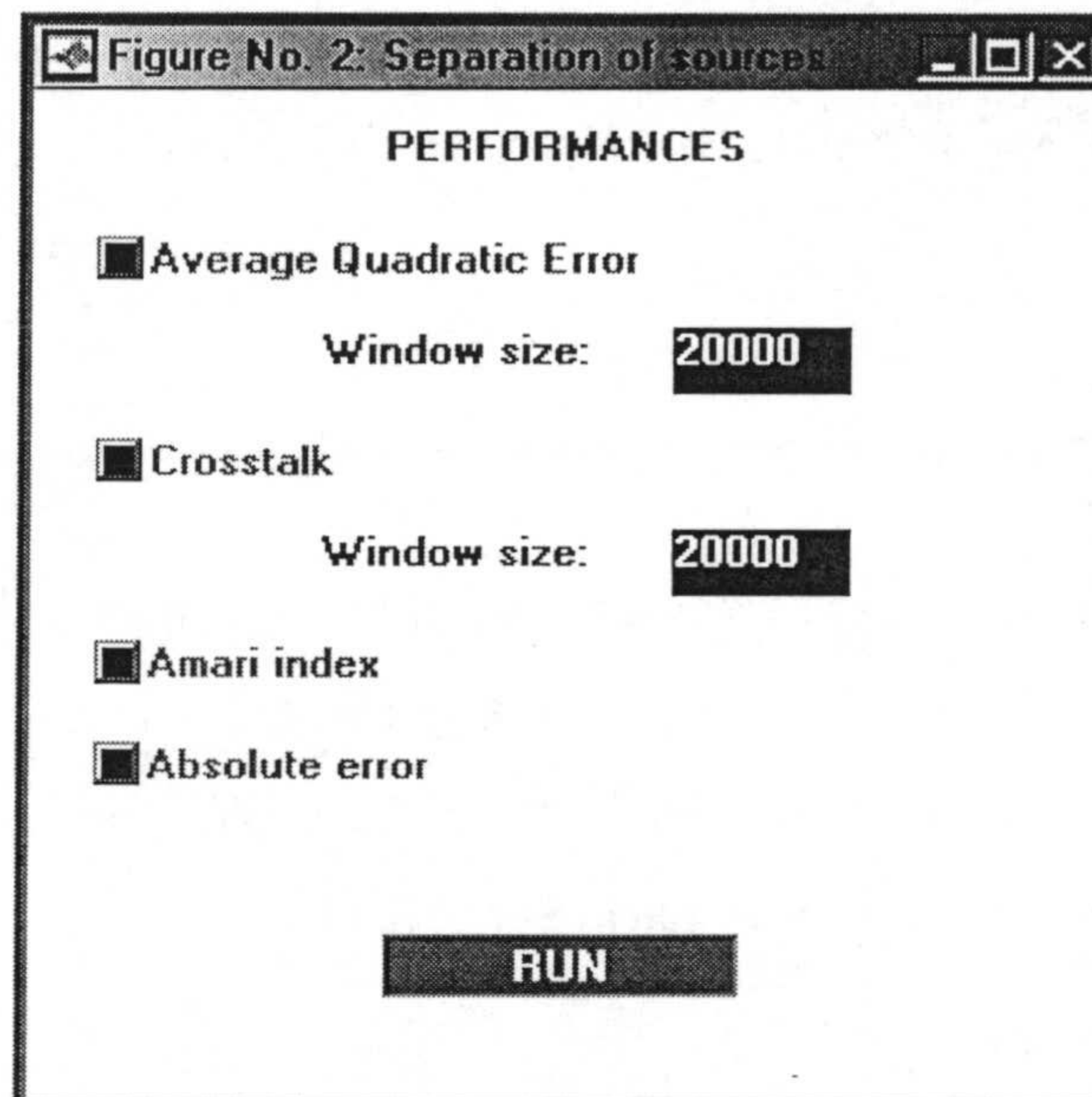


Figura 3.13. Medidas de las prestaciones.

El primer estimador es el error medio cuadrático, definido como (Sec.2.5.2.2):

$$RMS_{i(v)}(t) = \sum_{\tau=\tau-T \cdot N_v}^t \sqrt{\frac{[s_i(\tau) - y_i^*(\tau)]^2}{N_v}} \quad (3.13)$$

donde y_j^* representa las señales reconstruidas, pero escaladas de acuerdo con el rango de las señales originales, $s_i(t)$, y T es el periodo de muestreo. El cálculo del RMS se hace dentro de una ventana deslizante de N_v muestras (por defecto se toman 20.000); es decir, el valor RMS en un instante dado t se calcula considerando las últimas N_v muestras.

La segunda medida es la diafonía (“Crosstalk”, Sec.2.5.3), que de forma similar al RMS, se calcula dentro de una ventana deslizante (también por defecto de 20.000 muestras), y viene definido por:

$$C_{i(v)}(t) = 10 \cdot \log \cdot \left\{ \frac{\sum_{\tau=N_v}^t (s_i(\tau) - y_i^*(\tau))^2}{\sum_{\tau=N_v}^t (s_i(\tau))^2} \right\} \quad (dBs) \quad (3.14)$$

Otra posibilidad de comparación entre las señales recuperadas y las fuentes originales que ofrece el simulador es el módulo del error absoluto instantáneo:

$$\varepsilon_i(t) = |s_i(t) - y_i^*(t)| \quad (3.15)$$

Las medidas anteriores necesitan conocer las fuentes reconstruidas (y_i). Aunque estas medidas son intuitivas y fáciles de interpretar, con frecuencia resulta más cómodo utilizar un estimador que compare directamente las matrices \mathbf{A} y \mathbf{W} , para ello el simulador utiliza el índice de prestaciones propuesto por Amari (Sec.2.5.1):

$$PIA \stackrel{def}{=} \sum_{i=1}^p \left(\sum_{j=1}^p \frac{|p_{ij}|}{\max_k |p_{ik}|} - 1 \right) + \sum_{j=1}^p \left(\sum_{i=1}^p \frac{|p_{ij}|}{\max_k |p_{kj}|} - 1 \right) \quad (3.16)$$

donde $\mathbf{P}=(p_{ij})$ se definió en (2.10). Cuanto menor es PIA, más próximo está \mathbf{W} de satisfacer (3.4).

Con el menú que se muestra en la Figura 3.13 se puede seleccionar simultáneamente los índices de prestaciones que interesen.

En el caso del índice PIA aparece una ventana en la que se va viendo la evolución de dicho índice a lo largo del tiempo (Figura 3.14).

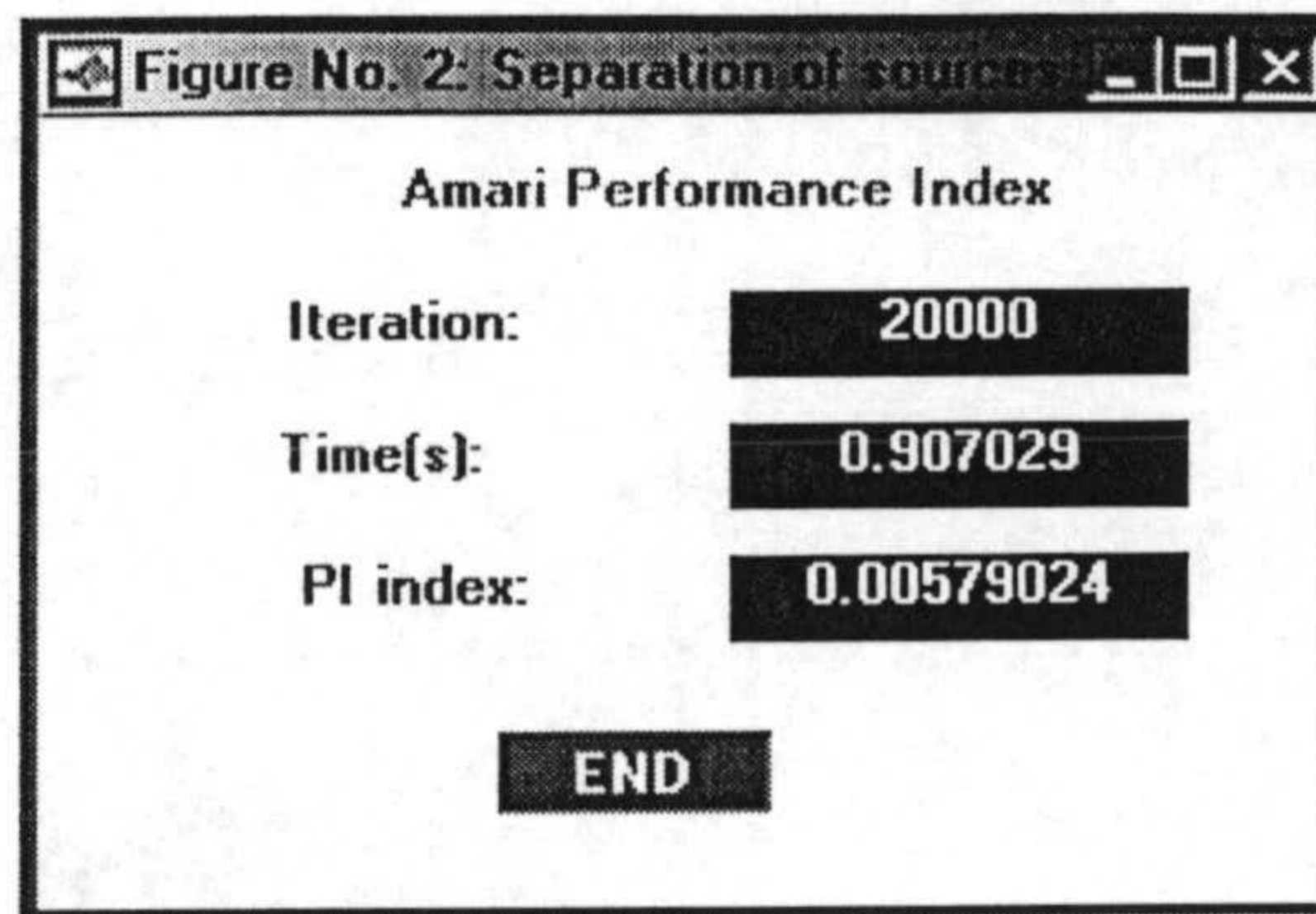


Figura 3.14. Evolución del índice de prestaciones de Amari.

3.8. MÓDULO DE RESULTADOS Y REPRESENTACIONES

El último módulo (Figura 3.15) permite visualizar en forma numérica y gráfica las señales involucradas en el proceso y los resultados obtenidos en las distintas fases. Las opciones del menú asociado a este módulo pueden elegirse entre cualquiera de las fases del proceso.

En cualquier momento se puede visualizar con gráficos:

- La evolución temporal (Figura 3.16a) tanto de las fuentes como de las mezclas y de las señales recuperadas (en la Figura 3.16b se ha representado la evolución temporal de las fuentes), con la posibilidad de cambiar la escala de los ejes (Figura 3.19).
- Los espacios vectoriales (Figura 3.17) de estas mismas señales. En la Figura 3.17b se muestra el espacio de mezclas.
- Las distribuciones de amplitudes (Figura 3.18). En la Figura 3.18b se puede ver la distribución de amplitudes de las fuentes.

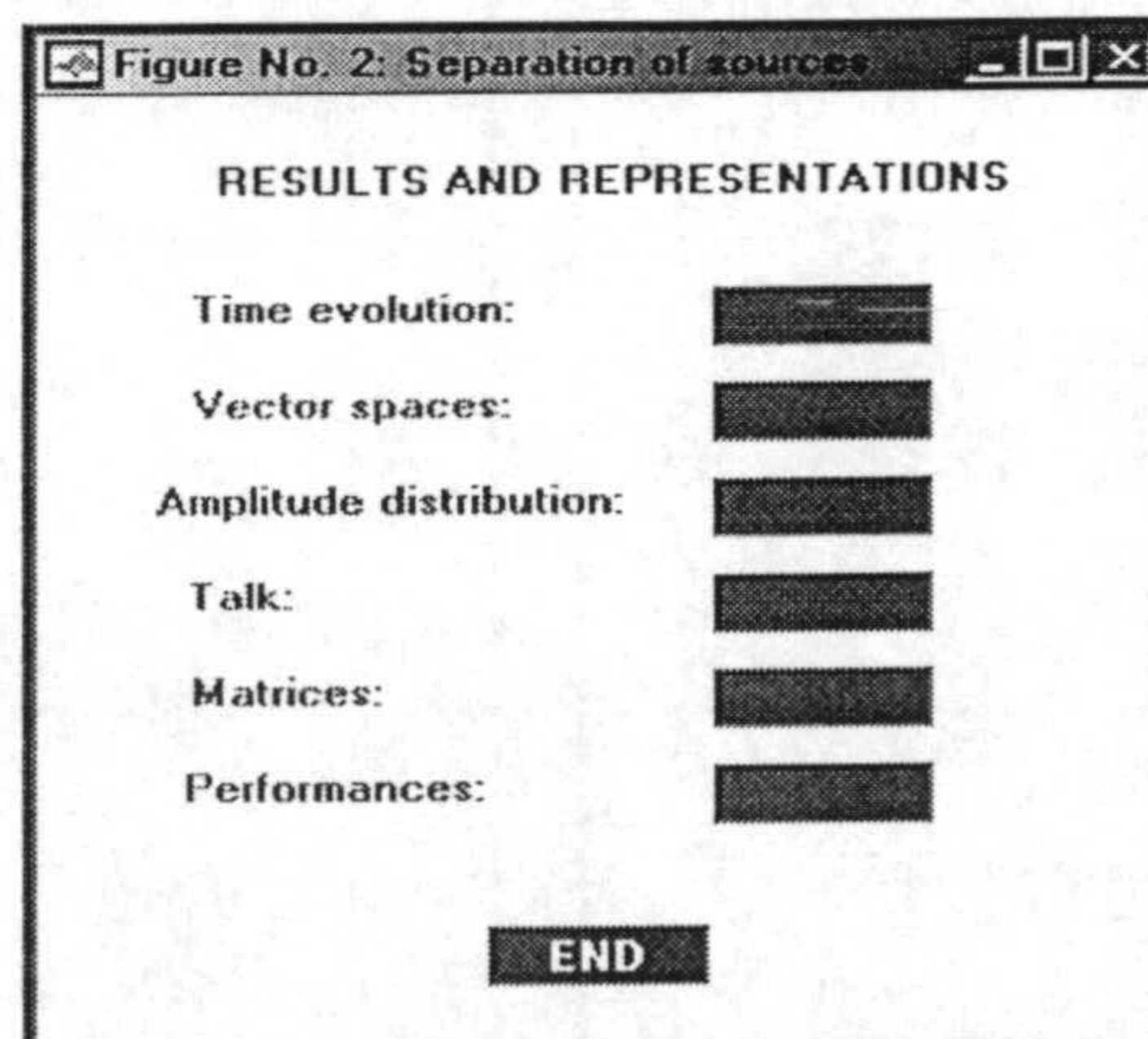
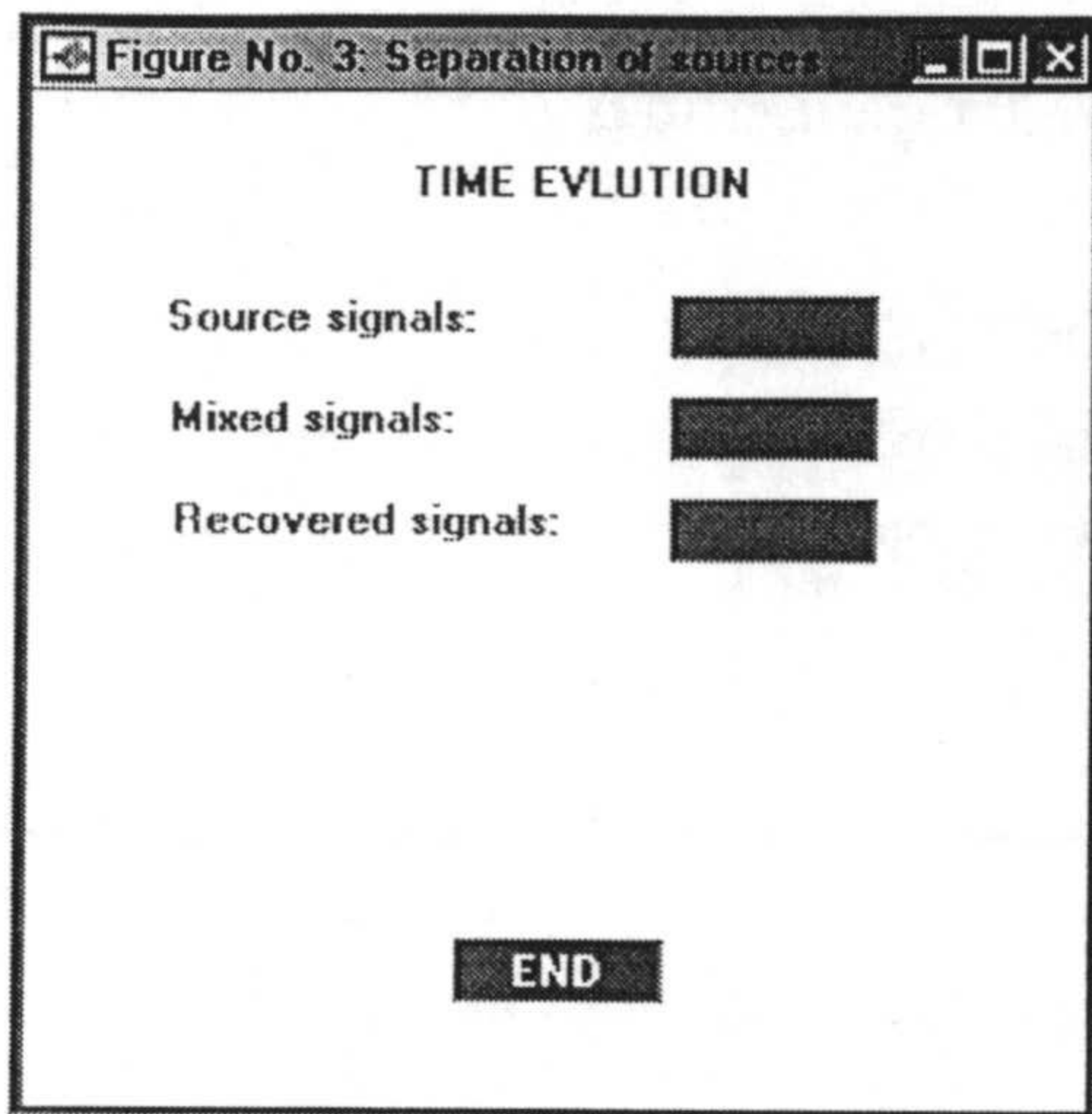
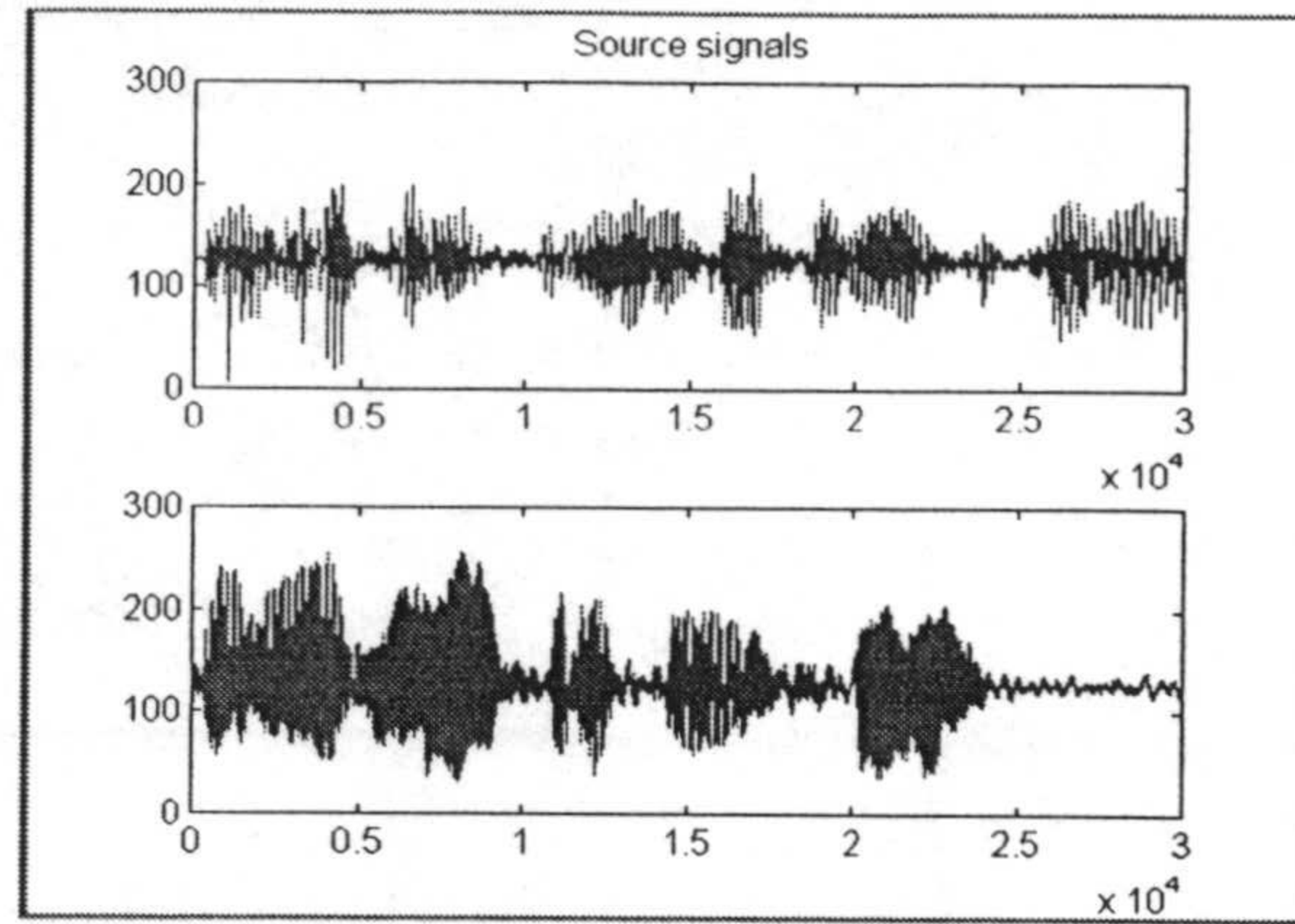


Figura 3.15. Módulo de resultados y representaciones.

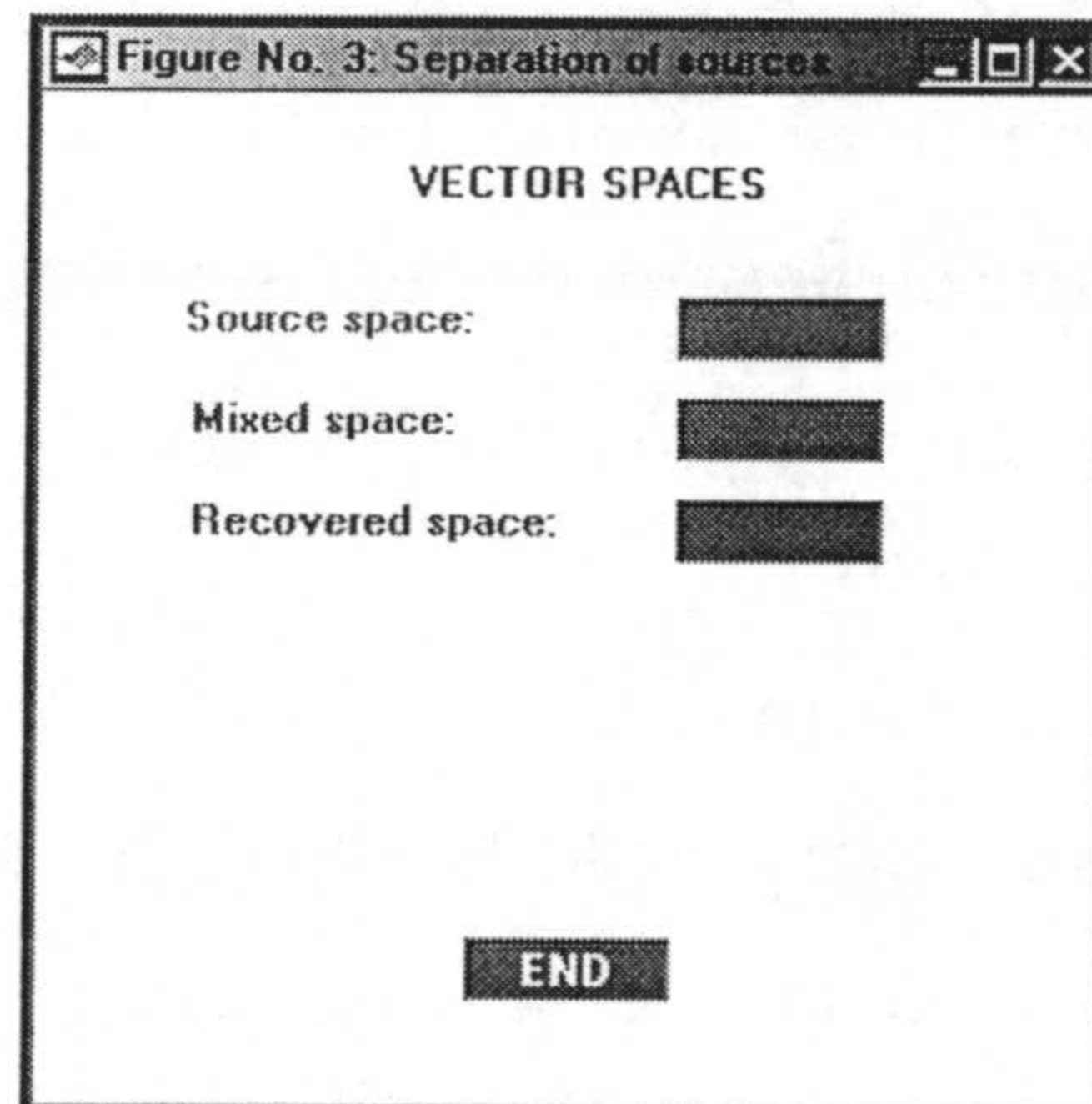


(a)

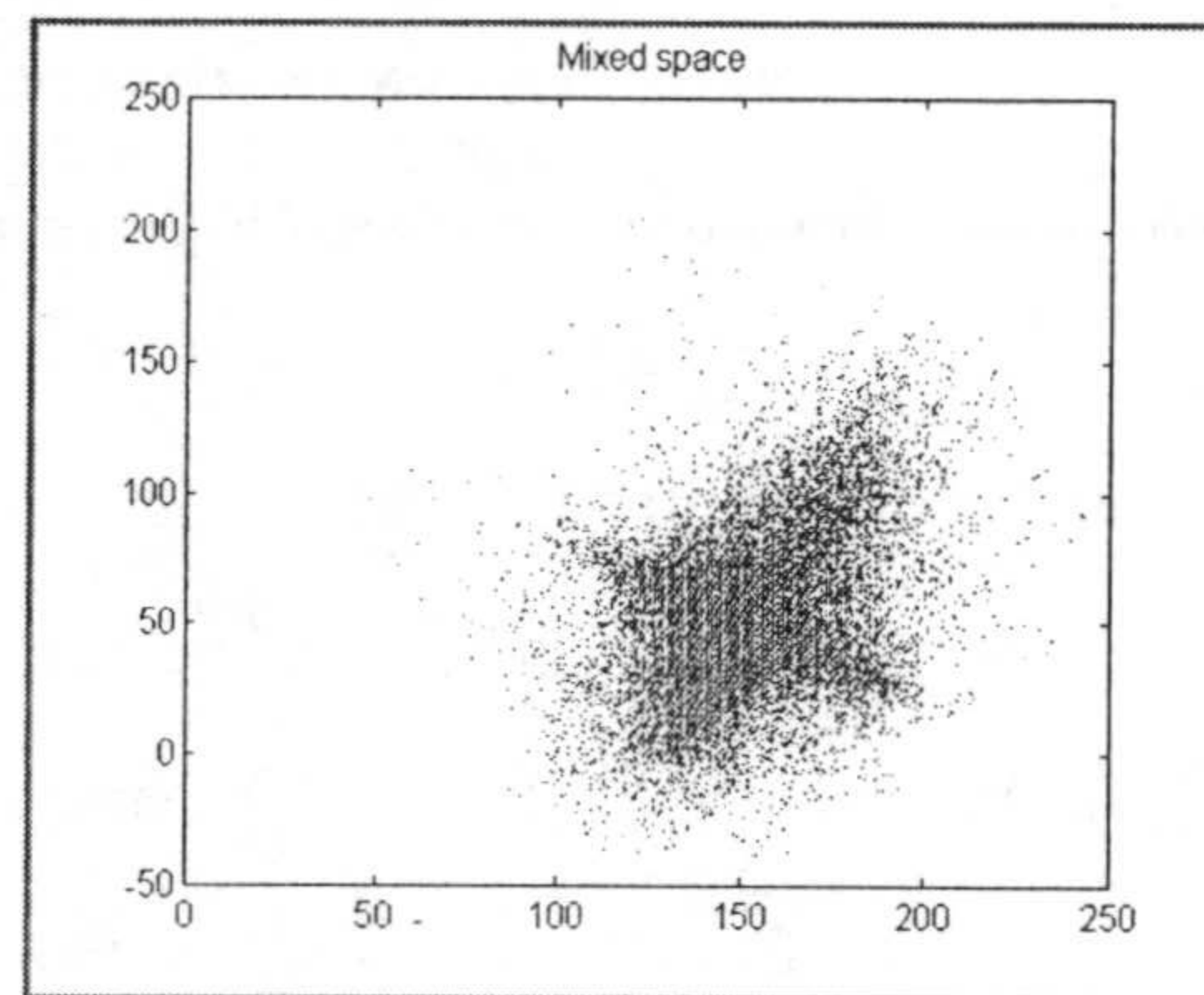


(b)

Figura 3.16. Evolución temporal.



(a)



(b)

Figura 3.17. Espacios vectoriales.

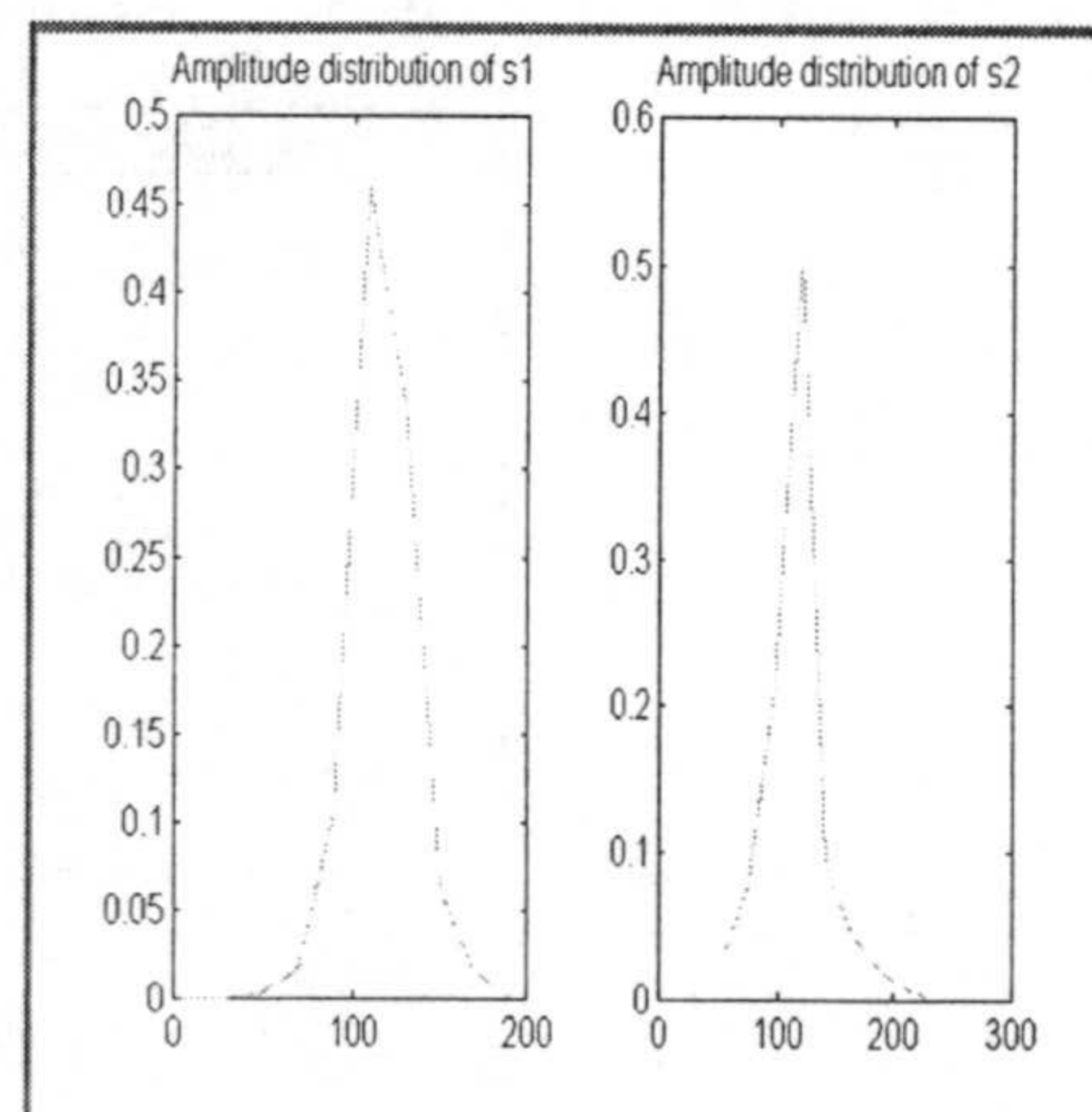
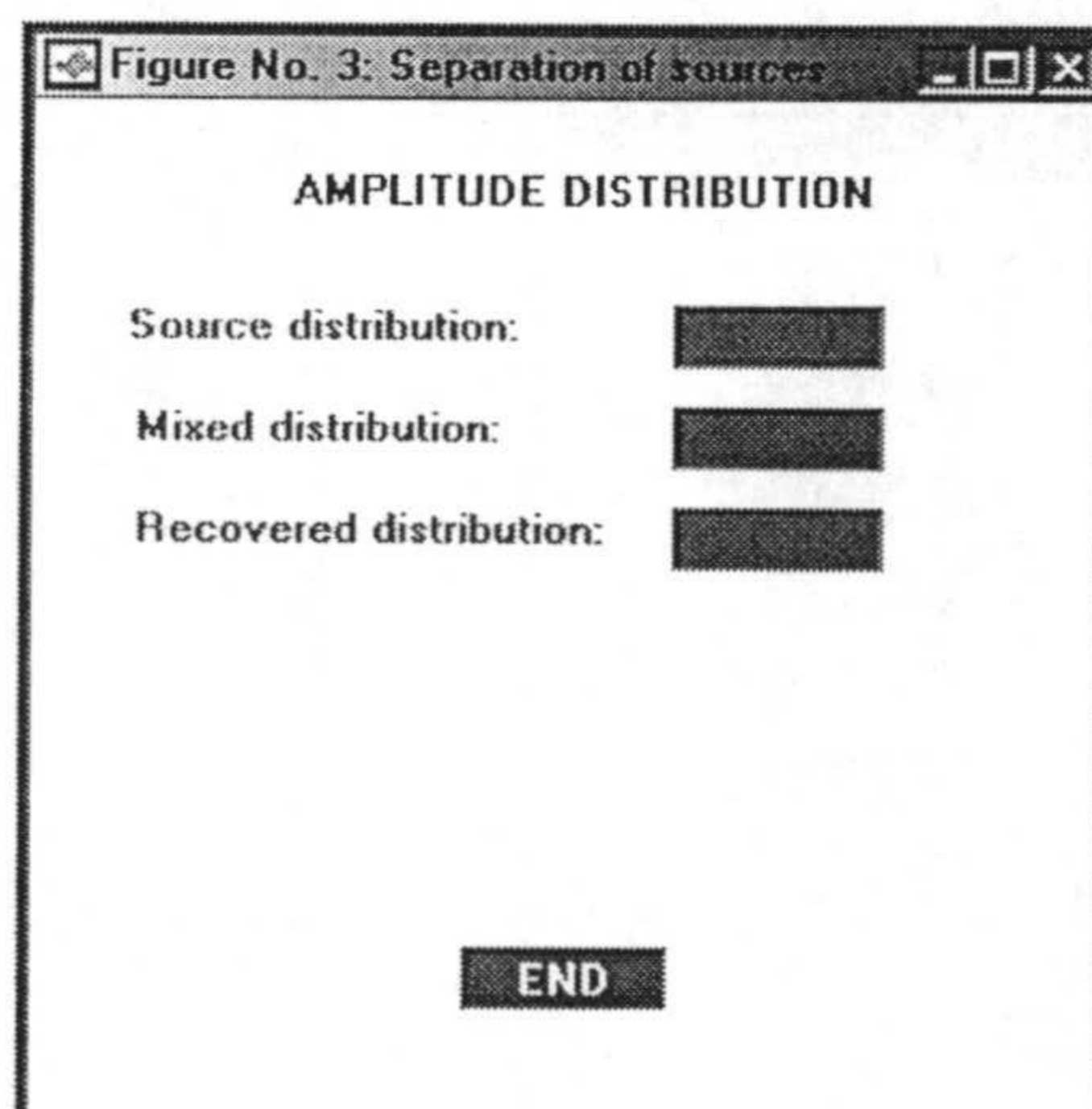


Figura 3.18. Distribución de amplitudes.

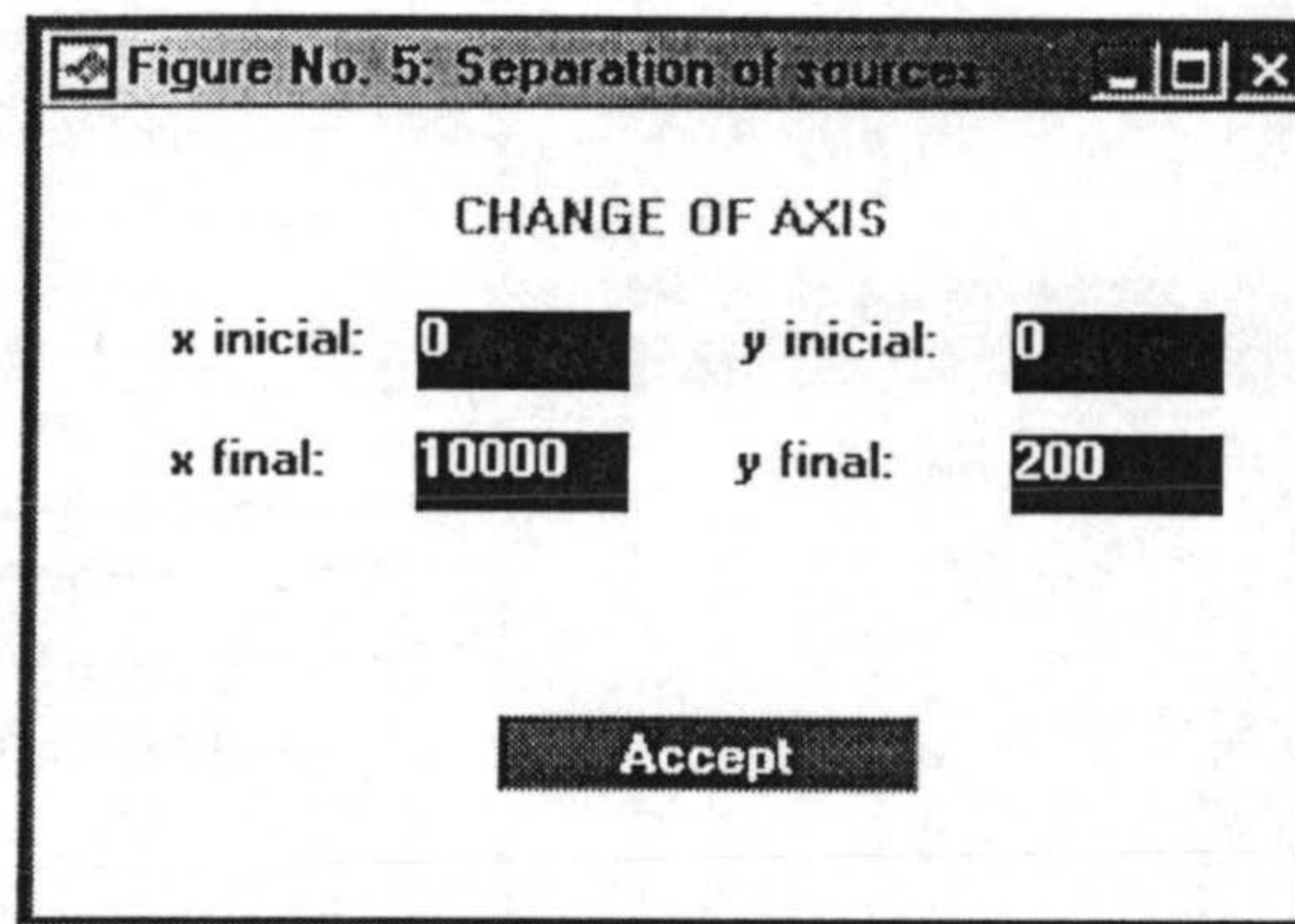


Figura 3.19. *Ventana para cambiar los ejes.*

Las demás opciones del menú permiten:

- Escuchar a través de los altavoces del PC o estación de trabajo las señales, ya sean las fuentes, las mezclas o las fuentes recuperadas, a través de los altavoces del sistema eligiéndolas una a una (Figura 3.20),

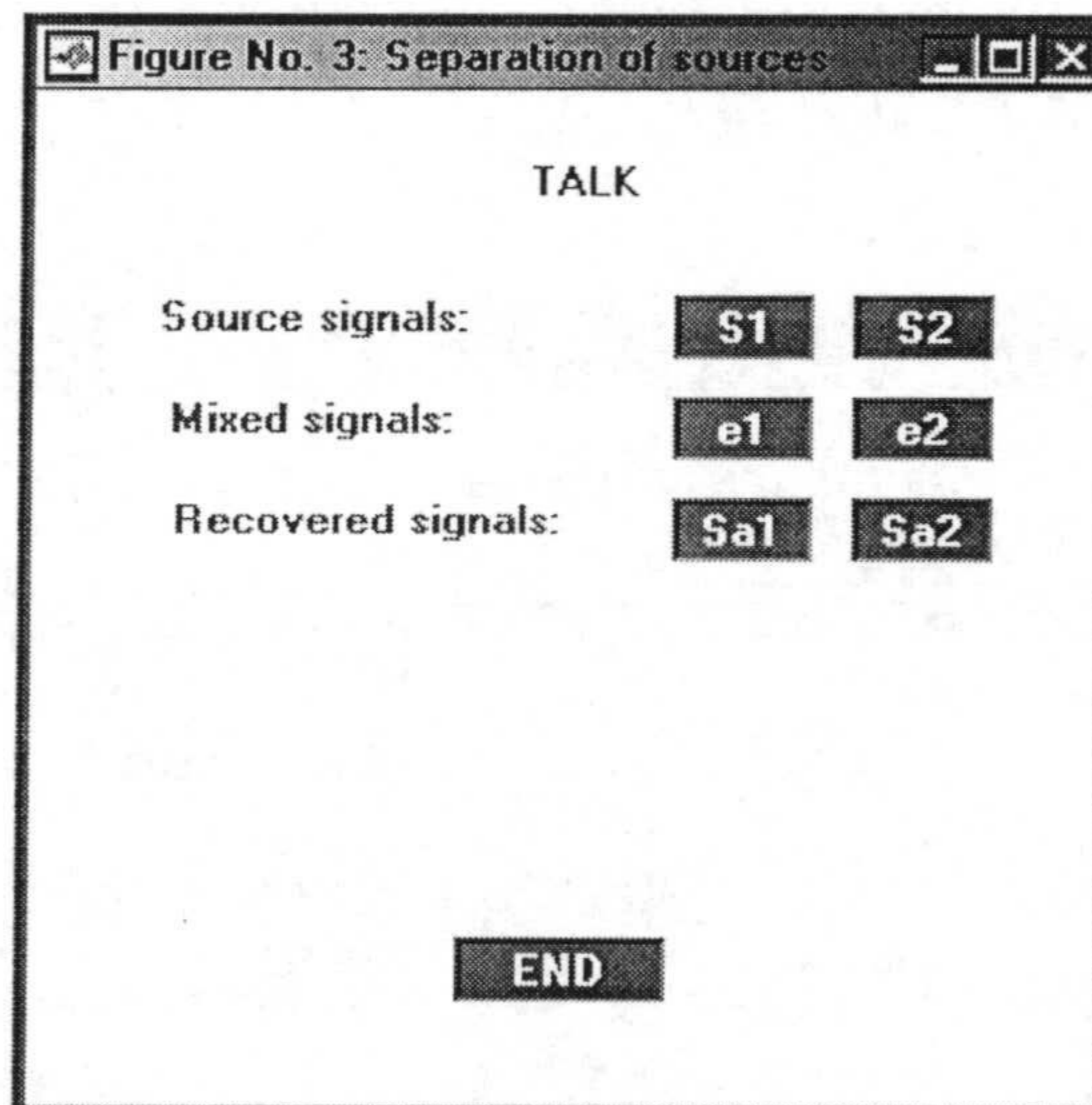


Figura 3.20. *Ventana para escuchar las señales.*

- Visualizar las matrices A , A canónica y W (matriz de mezcla recuperada) (Figura 3.21). Se visualizarán aquellas matrices que se seleccionen previamente, es decir, pueden ser todas, dos o una. Una vez seleccionadas, al pulsar el botón *show* aparecerán tantas ventanas como matrices se hallan seleccionadas, con sus respectivos valores.

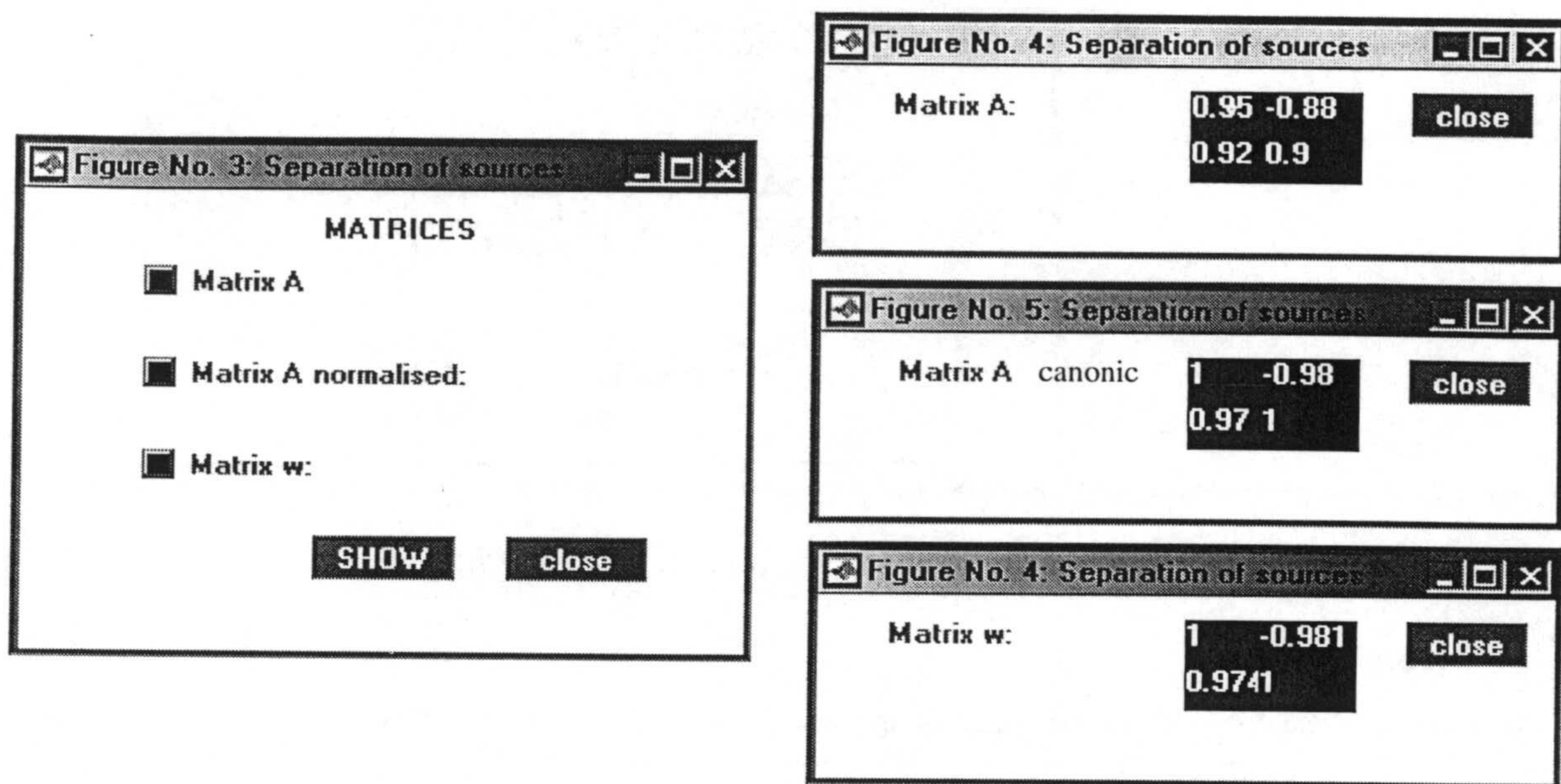


Figura 3.21. Visualización de las matrices A, Ad y W.

- Representar en el tiempo los distintos índices de prestaciones calculados en el módulo anterior (Sec.3.7) para lo cual el menú de la Figura 3.22 pide al usuario los parámetros cuya evolución se desea visualizar.

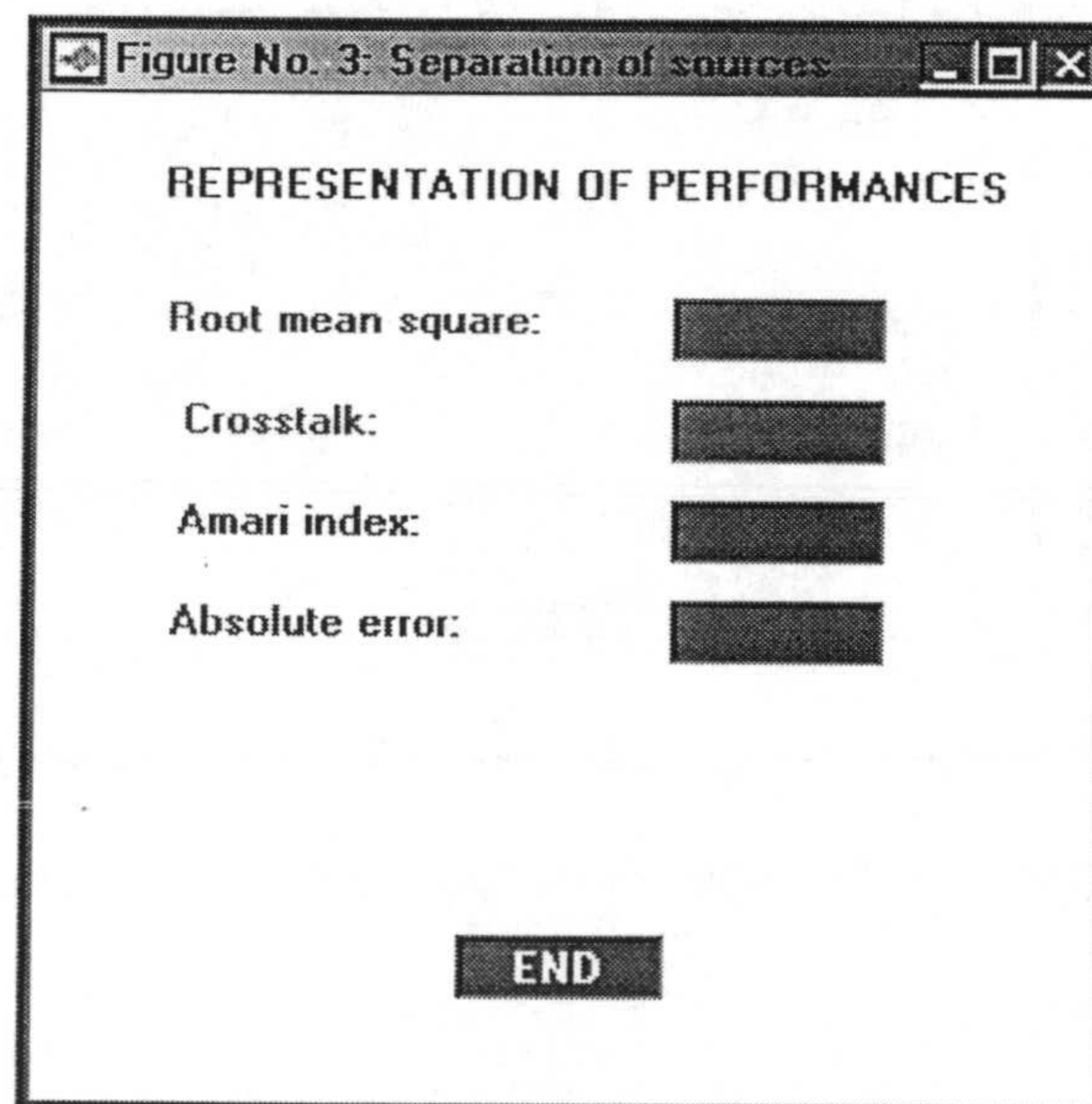


Figura 3.22. Módulo de representación de los índices de prestaciones.

En la Figura 3.23 se pueden ver las representaciones gráficas correspondientes a elegir cada opción del menú de representación de índices.

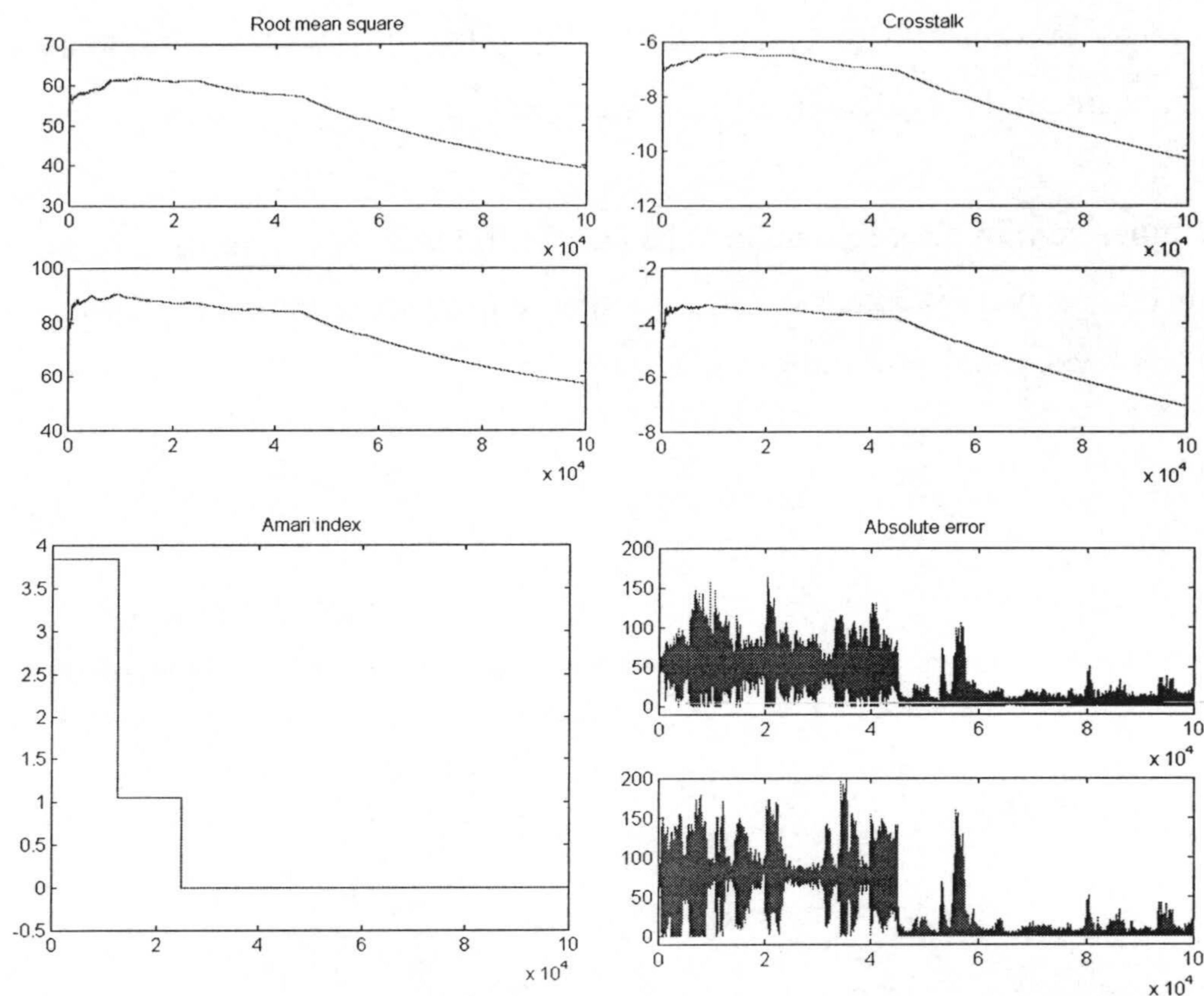


Figura 3.23. Gráficos obtenidos a partir del módulo de representación de los índices de prestaciones.

3.9. CONCLUSIONES

Este capítulo ha planteado el diseño e implementación de un entorno de simulación para facilitar el desarrollo de algoritmos para separación ciega de fuentes. La herramienta dispone de una biblioteca de señales, tanto sintéticas como reales, y permite simular la mezcla eligiendo, bien arbitrariamente por el usuario, o bien aleatoriamente, los elementos de la matriz de mezcla. El conjunto de programas incluye dos algoritmos para la estimación de la matriz de mezcla, y tres procedimientos para recuperar las señales originales partiendo de las señales de mezcla y de la matriz de mezcla estimada. El simulador también implementa distintos índices para poder medir la calidad de los resultados obtenidos: error absoluto, RMS, diafonía e índice de prestaciones de Amari.

En cualquier momento pueden visualizarse las distintas señales y resultados involucrados en el proceso. Las señales se visualizan como frecuencia de amplitudes, evolución en el tiempo y en forma de vectores. Este simulador puede utilizarse como herramienta para probar distintos algoritmos de separación ciega de fuentes.

Los programas se han desarrollado en MATLAB, la interfaz con el usuario es interactiva y basada en menús, y la modularidad del sistema completo permite ampliarlo con facilidad, incluyendo nuevas funciones y algoritmos.

Este entorno de simulación se ha usado para desarrollar y probar los algoritmos que se describen en los Capítulos 4 y 5.

CAPÍTULO 4:

ALGORITMO DE MÁXIMA SEPARACIÓN ANGULAR PARA SEPARACIÓN CIEGA DE FUENTES.

En este capítulo se presenta un algoritmo basado en propiedades geométricas (Capítulo 2) para separar mezclas lineales e instantáneas de p señales.

El algoritmo se basa en la utilización de dos redes neuronales artificiales, cuyos pesos son comunes y estiman los coeficientes de la mezcla. La red de recuperación es lineal de tipo recursivo y se describió en la Sec.3.6. La red de aprendizaje es de tipo competitivo (no supervisado) con p neuronas también lineales y otra de salida no lineal (función escalón) modificándose durante el aprendizaje tanto los pesos como el umbral de la función escalón. El aprendizaje y la separación se llevan a cabo muestra a muestra, no existiendo fases diferenciadas de aprendizaje y recuperación de fuentes.

El algoritmo converge más rápidamente cuanto más suprauniformes (Sec.A1.3.1) sean las funciones de distribución de las fuentes originales; ya que esto da lugar a una mayor

probabilidad de obtener puntos en los ejes del hiperparalelepípedo del espacio de observaciones.

En primer lugar (Sec.4.1) se describen los principios teóricos del método. Para aplicar el procedimiento es necesario efectuar una traslación del hiperparalelepípedo de observaciones según se indica en la Sec.4.2. En la Sec.4.3 se analiza la influencia del ruido y como reducir sus efectos en los resultados. La Sec.4.4 plantea el problema de la dependencia de la convergencia del método con la obtención de determinadas combinaciones de valores de las señales de entrada (*vectores críticos*). La Sec.4.5 describe la implementación del algoritmo por medio de dos redes neuronales: una de recuperación (Sec.4.5.1) y otra de aprendizaje (Sec.4.5.2). La Sec.4.6 da una visión general sobre la implementación realizada y detalla la fase de preproceso, tendente a reducir los efectos del ruido. En la Sec.4.7 se muestran los resultados experimentales obtenidos utilizando el simulador descrito en el Capítulo 3, tanto con fuentes sintéticas como reales. Finalmente la Sec.4.7 contiene una discusión de los resultados obtenidos y las principales conclusiones del capítulo.

4.1 PRINCIPIOS DEL MÉTODO

El procedimiento que se describe en este capítulo parte de las hipótesis descritas y justificadas en la Sec.2.1.1.

En particular, la Hipótesis 2.3 establece que las fuentes están acotadas, y como consecuencia de ello el espacio de observaciones conforma un hiperparalelepípedo (Sec.2.2). En principio supóngase que este hiperparalelepípedo tiene un vértice en el origen de forma tal que $s_i(t) \in [0, s_{iM}]$, $s_{iM} \in \mathcal{R}$.

El algoritmo que se propone en este capítulo trata de identificar, entre los vectores observados, aquéllos que se encuentren en las aristas del cono del hiperparalelepípedo que contienen el espacio de observaciones, y cuyo vértice es el origen.

Recuérdese (Propiedad 2.6, Sec.2.3) que se puede estimar la matriz de mezcla encontrando p vectores (vectores arista) situados cada uno de ellos en uno de los ejes del

cono que contiene el hiperparalelepípedo de observaciones. El método propuesto en este capítulo se basa en que un conjunto de p vectores, con uno de sus extremos en el vértice del cono y con *separación angular* máxima constituyen un conjunto de vectores arista.

Definición 4.1. Proximidad angular, γ_{uv} , entre dos vectores $e(u)$, $e(v)$.

Como las componentes de dos vectores, $e(u)$ y $e(v)$, en los instantes u y v son conocidos, es fácil calcular el coseno del ángulo que forman, o *proximidad angular*, γ_{uv} , a partir de su producto escalar. Es decir, se puede evaluar la siguiente expresión:

$$\gamma_{uv} = \cos[e(u), e(v)] = \frac{\sum_{j=1}^p e_j(u)e_j(v)}{|e(u)| \cdot |e(v)|} = \frac{e(u) \cdot e(v)}{|e(u)| \cdot |e(v)|} \quad (4.1)$$

Según esta definición, la proximidad angular de dos vectores es máxima, $+1$, cuando los dos vectores coinciden (ángulo de 0°) y es mínima, -1 , cuando la separación sea máxima (ángulo de 180°).

Definición 4.2. Proximidad angular, γ_C , de un conjunto C de vectores.

La *proximidad angular*, γ_C , de un conjunto C , de p vectores, $C = \{e_1, \dots, e_p\}$, se define como la suma de los cosenos de los ángulos formados entre dichos vectores, es decir, es una aplicación $\gamma: \mathfrak{R}^{p \times p} \rightarrow \mathfrak{R}$ definida de forma que:

$$\gamma_C \equiv \sum_{u=1}^{p-1} \sum_{v=u+1}^p \cos[e(u), e(v)] = \sum_{u=1}^{p-1} \sum_{v=u+1}^p \frac{e(u) \cdot e(v)}{|e(u)| \cdot |e(v)|} \quad \text{con } v \neq u \quad (4.2)$$

En particular, según (4.4), la proximidad angular, γ_w , de un conjunto, C_w , de p vectores situados cada uno en una de las aristas del hiperparalelepípedo tendrá la forma siguiente:

$$\gamma_w \equiv \sum_{i=1}^{p-1} \sum_{j=i+1}^p \cos[w(i), w(j)] = \sum_{i=1}^{p-1} \sum_{j=i+1}^p \frac{w(i) \cdot w(j)}{|w(i)| \cdot |w(j)|} \quad \text{con } i \neq j \quad (4.3)$$

Teorema 4.1.

La proximidad angular de p vectores del espacio de observaciones es mínima (o sea, dichos vectores tendrán una separación angular máxima) si y sólo si forman un conjunto de p vectores arista, es decir, si verifican:

$$\gamma_w \leq \gamma_c \quad (4.4)$$

En otras palabras, los p vectores arista se pueden obtener cuando la proximidad, γ_c , tome su valor mínimo.

Demostración.

Consideremos una hiperesfera de radio r y con centro en uno de los vértices del cono del hiperparalelepípedo del espacio de observaciones, E . Esta hiperesfera cortará al hiperparalelepípedo en p puntos $\{p_1, \dots, p_p\}$ situados en las aristas de dicho hiperparalelepípedo. A partir del hiperplano que pasa por los puntos $\{p_1, \dots, p_p\}$ se forma un hiperpolígono (para $p=2$ será una línea, para $p=3$ será un triángulo, etc.) de vértices $\{p_1, \dots, p_p\}$. Por tanto, los vectores arista se pueden proyectar en un hiperpolígono cuyos vértices disten r del vértice del cono (0). Cada par de puntos (p_i, p_j) forma un plano $\{0, p_i, p_j\}$ con el vértice del cono (0). Sea L_{ij} la distancia entre p_i y p_j (Figura 4.1).

Por la ley de los cosenos:

$$L_{ij} = (r^2 + r^2 - 2r^2 \cos \theta_{ij})^{1/2} = \sqrt{2} \cdot r \cdot \sqrt{(1 - \cos \theta_{ij})} \quad (4.5)$$

en este caso se verifica que:

$$0 \leq \theta_{ij} < \pi \quad \text{ó} \quad -1 < \cos \theta_{ij} < 1 \quad (4.6)$$

y por tanto se puede hacer la siguiente aproximación:

$$\sqrt{1 - \cos \theta_{ij}} \approx 1 - \frac{1}{2} \cos \theta_{ij} \quad -1 \leq \cos \theta_{ij} < 1 \quad (4.7)$$

entonces (4.7) quedaría:

$$L_{ij} \approx \sqrt{2} \cdot r \cdot \left(1 - \frac{1}{2} \cdot \cos \theta_{ij} \right) \quad (4.8)$$

Cualquier conjunto de p vectores en el espacio de observaciones que tengan su origen en el vértice 0 se pueden proyectar, como se vio antes, en un hiperpolígono cuya área es máxima cuando los p vectores se encuentren en las aristas del hiperparalelepípedo. Por ejemplo, para $p=3$, el hiperpolígono es un triángulo y su área (K) dependerá del perímetro L_T y de la longitud de los lados L_{ij} :

$$K = \sqrt{L_T \prod_{i,j} (L_T - L_{ij})} \quad L_T = \sum_{i,j} L_{ij} \quad \begin{matrix} i = 1, \dots, p-1 \\ j = i+1, \dots, p \end{matrix} \quad (4.9)$$

En este caso también ($p=3$), el área externa es igual a la suma de las áreas formadas por los tres triángulos $(0, P_1, P_2)$, $(0, P_1, P_3)$ y $(0, P_2, P_3)$, más el área de cada triángulo que es proporcional a L_{ij} . Para p dimensiones, existen $\frac{p(p-1)}{2}$ triángulos $(0, P_i, P_j)$, y por tanto, el área externa (K') que forman será proporcional a L_{ij} , ya que $K' = K \frac{p(p-1)}{2}$. Por consiguiente, en general (p dimensiones) el área será máxima cuando la longitud L_{ij} sea máxima. Es decir, cuando la longitud total L_T , sea máxima. Teniendo en cuenta (4.10):

$$L_T = \sum_{i=1}^{p-1} \sum_{j=i+1}^p L_{ij} \approx p\sqrt{2} \cdot r - \frac{\sqrt{2}}{2} \cdot r \cdot \left(\sum_{i=1}^{p-1} \sum_{j=i+1}^p \cos \theta_{ij} \right) \quad (4.10)$$

ó considerando γ_C :

$$\gamma_C = \sum_{i=1}^{p-1} \sum_{j=i+1}^p \cos \theta_{ij} \approx 2p - \frac{\sqrt{2}}{2} \cdot L_T \quad (4.11)$$

A partir de (4.11) se puede deducir que para un p dado, si L_T es máximo, la separación angular γ_C será mínima. Como L_T es máximo si y sólo si los p puntos pertenecen a los vectores arista, se verifica que:

$$\gamma_w \leq \gamma_c \quad (4.12)$$

con lo que queda demostrado el Teorema 4.1.

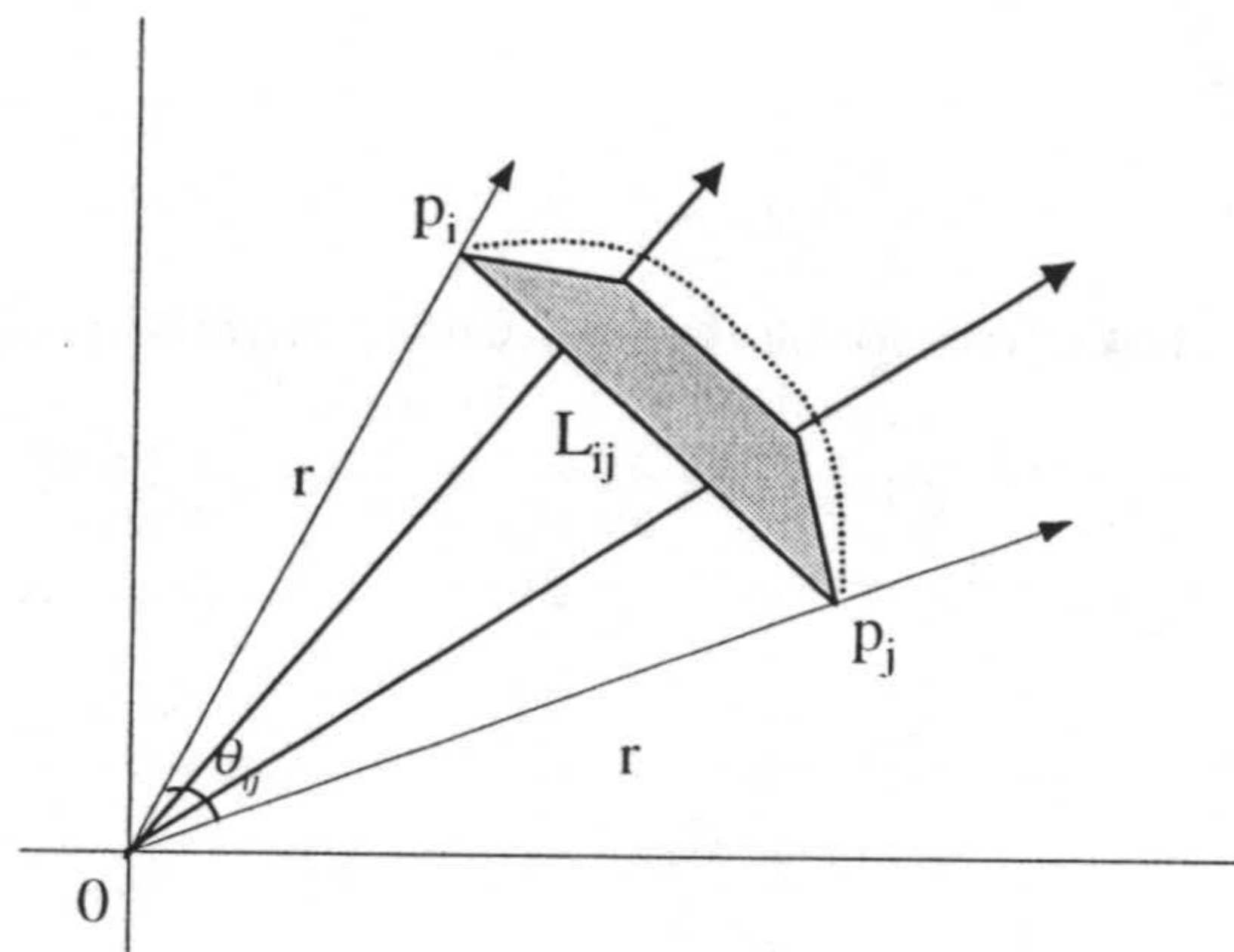


Figura 4.1. Hiperpolígono del espacio de observación.

El proceso iterativo que se propone como método de separación considera los p primeros vectores observados y forma con ellos un supuesto conjunto, C_w , de vectores arista, y con ellos calcula su proximidad angular, γ_w . Cuando llega un nuevo vector, e , dicho vector se añade al conjunto, formando ahora un conjunto de test, C_t , que contiene $p+1$ vectores. El subconjunto C de C_t , de p vectores, que haga mínimo γ_c , se toma como nuevo hipotético conjunto de vectores arista, C_w . Así, al final del proceso iterativo se tendrá un conjunto de p vectores cuya proximidad será mínima (separación angular máxima) y que corresponderá a los vectores arista del hiperparalelepípedo de observación.

4.2 TRASLACIÓN DEL HIPERPARALELEPÍPEDO.

Está claro, desde un punto de vista geométrico y teniendo en cuenta la linealidad de la matriz A , que el caso más general, $s_i(t) \in [s_{im}, s_{iM}]$ con $s_{im}, s_{iM} \in \mathcal{R}$, $s_{im} < s_{iM}$, equivale a la situación anterior después de efectuar un cambio de coordenadas con un vector ν , que representa un vector formado por las coordenadas de uno de los vértices del hiperparalelepípedo. Este cambio de coordenadas es equivalente a trasladar el hiperparalelepípedo de forma que el vértice ν se sitúe en el origen de coordenadas. En el caso de $p=3$, se tendría que hacer el cambio indicado en la Figura 4.2. De esta forma el problema se transforma en un caso $s_i \in [0, s_{iM}]$. Para llevar a cabo dicha traslación, el

algoritmo tiene que detectar el vector v . Para ello tiene en cuenta que dicho vector será el vector del espacio de observaciones cuyas componentes sean las coordenadas de cualquiera de los vértices. Este vértice se puede obtener de acuerdo con el procedimiento indicado en la Sec.2.4.

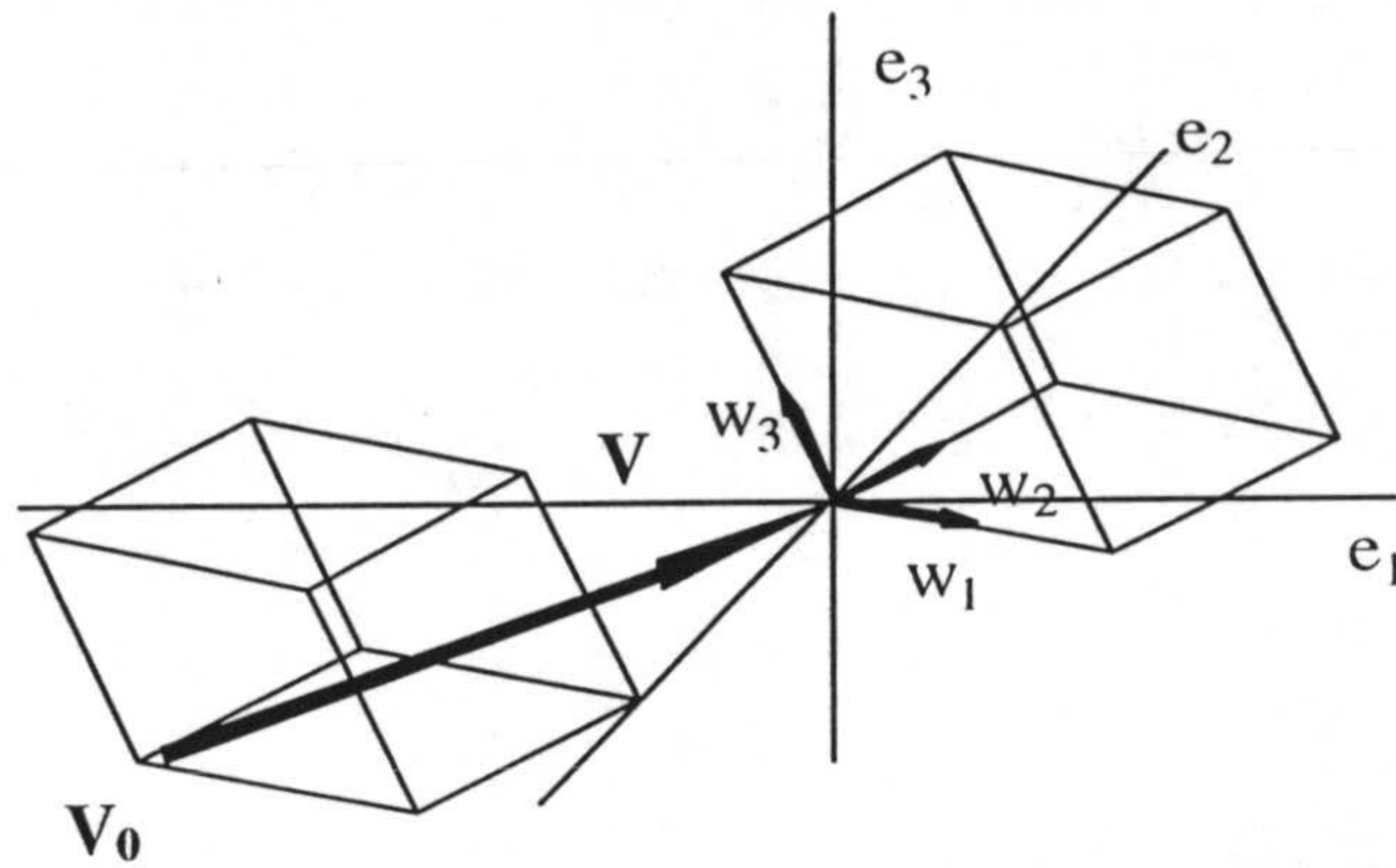


Figura 4.2. *Traslación del hiperparalelepípedo.*

En el caso de tres fuentes, se tendrían que detectar uno de los 8 vértices del hiperparalelepípedo. Recuerdese que según la Propiedad 2.7 (Sec.2.3) cualquier vértice puede utilizarse como referencia. Supóngase que el hiperparalelepípedo tiene inicialmente los siguientes vértices:

$$\begin{array}{ll}
 \mathbf{V}_0 = (-5,10,-11) & \mathbf{V}_4 = (-11,3,-3) \\
 \mathbf{V}_1 = (3,10,-10) & \mathbf{V}_5 = (-4,2,-2) \\
 \mathbf{V}_2 = (4,6,-6) & \mathbf{V}_6 = (-5,-4,4) \\
 \mathbf{V}_3 = (-4,6,-7) & \mathbf{V}_7 = (-12,-3,3)
 \end{array} \tag{4.13}$$

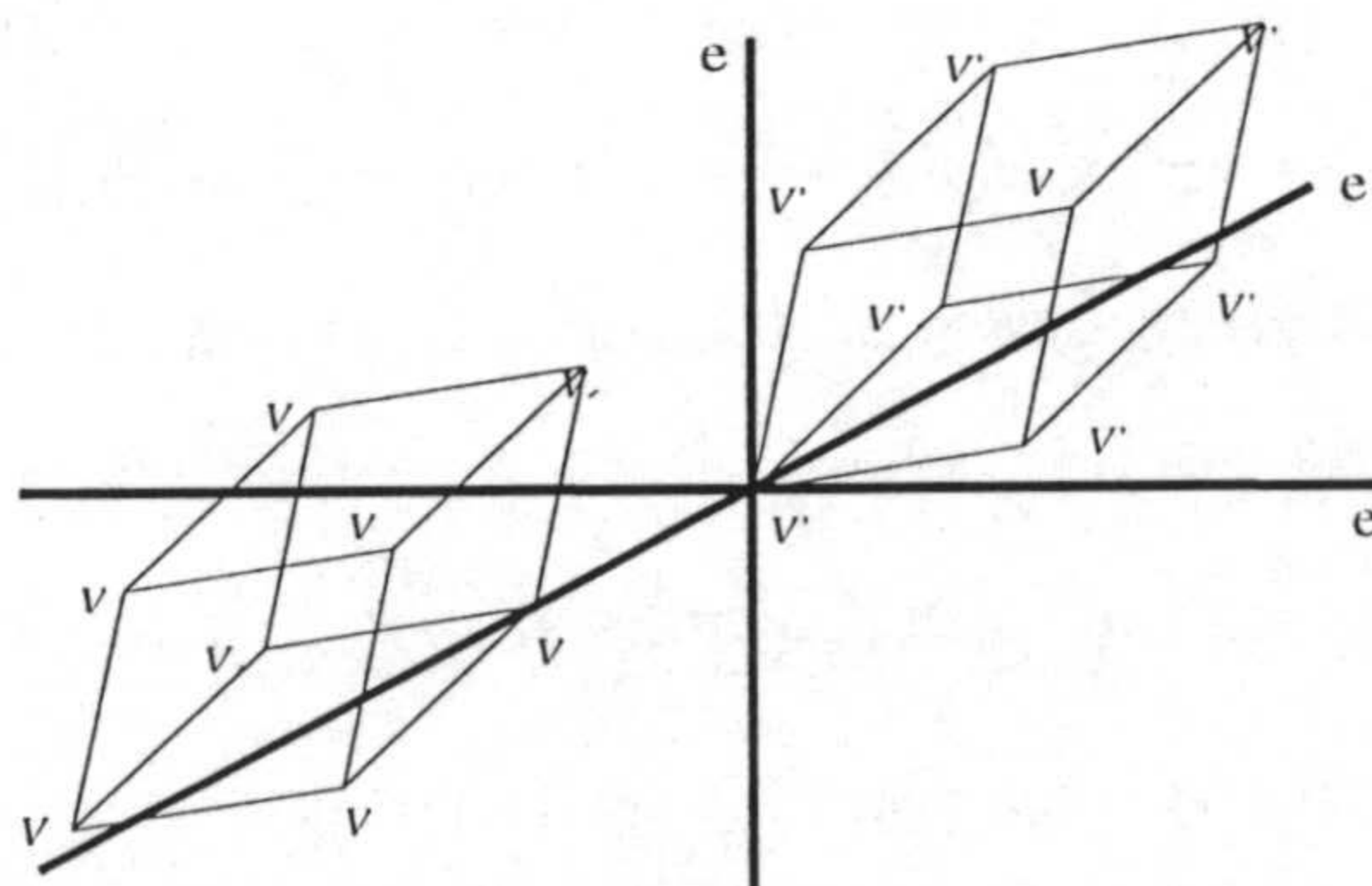


Figura 4.3. *Traslación en un caso $p=3$.*

Todos los vectores de la mezcla están contenidos en el hiperparalelepípedo (Figura 4.3). Se puede utilizar, por ejemplo, como vértice de traslación aquél, de entre todos los vectores

observados, cuya componente e_3 sea mínima. Puede observarse que, efectivamente, la componente e_3 de \mathbf{V}_0 es la menor de todos los vértices. Esto quiere decir que tenemos que efectuar el cambio de coordenadas, o traslación, con el vector ($\mathbf{v} = -\mathbf{V}_0$). De esta forma, los vectores del hiperparalelepípedo una vez trasladados (Figura 4.3) serían:

$$\begin{aligned}
 \mathbf{V}'_0 &= \mathbf{V}_0 - \mathbf{V}_0 = (0,0,0) & \mathbf{V}'_4 &= \mathbf{V}_4 - \mathbf{V}_0 = (-6,-7,8) \\
 \mathbf{V}'_1 &= \mathbf{V}_1 - \mathbf{V}_0 = (8,0,1) & \mathbf{V}'_5 &= \mathbf{V}_5 - \mathbf{V}_0 = (1,-8,9) \\
 \mathbf{V}'_2 &= \mathbf{V}_2 - \mathbf{V}_0 = (9,-4,5) & \mathbf{V}'_6 &= \mathbf{V}_6 - \mathbf{V}_0 = (0,-14,15) \\
 \mathbf{V}'_3 &= \mathbf{V}_3 - \mathbf{V}_0 = (1,-4,4) & \mathbf{V}'_7 &= \mathbf{V}_7 - \mathbf{V}_0 = (-7,-13,14)
 \end{aligned}
 \tag{4.14}$$

4.3 SEÑALES CON RUIDO

Tal como se indicó en la Sec.2.1.2, el ruido puede afectar de forma diferente a las señales originales o a la mezcla, y por tanto su tratamiento variará en función de dicha afectación.

El algoritmo propuesto en la Sección 4.1 permite separar fuentes mezcladas linealmente que pueden ser tanto deterministas como aleatorias. Por tanto, el ruido introducido en el medio por una fuente externa puede considerarse como una fuente más. En otras palabras, $s_i(t)$ puede ser: $s_i(t) = n_i(t)$, donde $n_i(t)$ es una señal de ruido. En la Sec.4.7 (Ejemplo 1) se incluye un ejemplo práctico de esta situación con señales reales y ruido (uniforme).

Otro caso consiste en que las señales originales, $s(t)$, hayan sido generadas con cierto ruido. En esta ocasión, el algoritmo propuesto es transparente a dicho ruido, recuperándose las señales con igual relación señal ruido y sin tener que modificar el algoritmo. Esto se mostrará experimentalmente en la Sec.4.7 (Ejemplo 3) donde se muestran los resultados de simulación de separación de señales reales con una relación señal/ruido de 24 dBs. Los resultados obtenidos demuestran las prestaciones satisfactorias del algoritmo.

Una tercera situación tiene lugar cuando se suma, dentro del medio, a las señales mezcladas un ruido, $n(t)$. En este caso se dice que el ruido corrompe de forma aditiva a las señales. Este efecto se puede modelar de la siguiente forma:

$$e^{(c)}(t) = A \cdot s(t) + n(t) = e(t) + n(t) \quad (4.15)$$

donde $n(t)$ usualmente se modela con una función de distribución de tipo uniforme, gaussiana, o laplaciana. En este caso, en general, hay que eliminar o filtrar las componentes $n(t)$ de la mezcla corrompida, $e^{(c)}(t)$, antes de llevar a cabo el proceso de separación. Para ello, se pueden usar técnicas convencionales de eliminación de ruido [MIT93]. Sin embargo, el algoritmo geométrico propuesto ofrece una cierta inmunidad al ruido como se ve a continuación.

Supóngase que el ruido $n_k(t)$, que afecta a cada señal $e_k(t)$, está acotado, es decir:

$$n_k(t) \subset [n_{km}, n_{kM}] \quad \text{con} \quad \begin{array}{l} n_{km} < n_{kM} \\ n_{km} < 0 \quad n_{kM} > 0 \end{array} \quad k = 1, \dots, p \quad (4.16)$$

En particular, el ruido blanco satisface (4.16). En estas condiciones, los vectores arista se corresponden con la zona sombreada de la Figura 4.4. El algoritmo, como se dijo en la Sección 4.1, detecta el vector (corrompido por el ruido) más pequeño, $e_0^{(c)}$, que se relaciona con el vector sin ruido, e_0 , de la siguiente forma:

$$e_0^{(c)} = e_0 + \mathbf{n}_m \quad (4.17)$$

donde $\mathbf{n}_m = (n_{km})$ representa la cota inferior de los vectores ruido, a partir de la que hay que obtener el vértice del hiperparalelepípedo.

Considérense dos aristas correlativas, i y j , del hiperparalelepípedo tal como muestra la Figura 4.4, donde todos los vectores representan sus proyecciones en el plano (i, j) . Los puntos de mayor separación angular estarán situados en las zonas sombreadas que representan las secciones donde se pueden encontrar los vectores arista corrompidos por el ruido. Los límites de estas zonas sombreadas corresponden a los puntos de las aristas con valores extremos de error $(\mathbf{n}_m, \mathbf{n}_M)$. Teniendo en cuenta el cambio de coordenadas, los vectores arista con menor proximidad angular serían:

$$\begin{aligned} \mathbf{w}^{(i)(c)} &= \mathbf{e}^{(i)(c)} - \mathbf{e}_0^{(c)} = (\mathbf{e}^{(i)} + \mathbf{n}_{ix}) - (\mathbf{e}_0 + \mathbf{n}_{ix}) = \mathbf{w}^{(i)} + (\mathbf{n}_{ix} - \mathbf{n}_m) \\ \mathbf{w}^{(j)(c)} &= \mathbf{e}^{(j)(c)} - \mathbf{e}_0^{(c)} = (\mathbf{e}^{(j)} + \mathbf{n}_{jx}) - (\mathbf{e}_0 + \mathbf{n}_{ix}) = \mathbf{w}^{(j)} + (\mathbf{n}_{jx} - \mathbf{n}_m) \end{aligned} \quad (4.18)$$

donde \mathbf{n}_{ix} y \mathbf{n}_{jx} representan el ruido en los puntos $\mathbf{e}^{(i)}$ y $\mathbf{e}^{(j)}$ respectivamente.

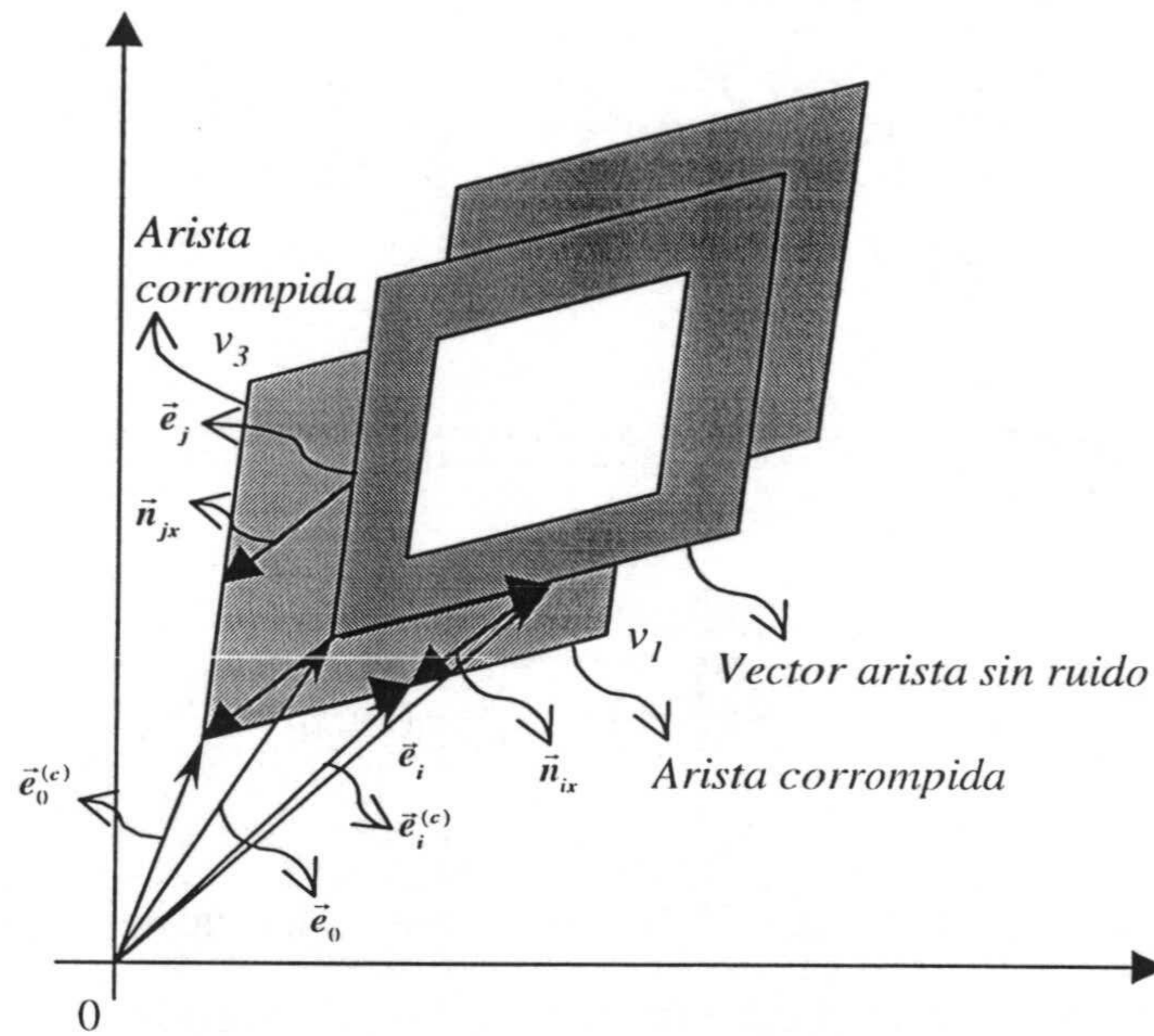


Figura 4.4. Influencia del ruido.

A partir de (4.18) se puede deducir que:

1. Cuando se detectan los vectores arista corrompidos en el nivel inferior de ruido ($\mathbf{n}_{ix} = \mathbf{n}_{jx} = \mathbf{n}_m$) el ruido se compensa, y según (4.18) se verificaría que $\mathbf{w}^{(i)(c)} = \mathbf{w}^{(i)}$, $\mathbf{w}^{(j)(c)} = \mathbf{w}^{(j)}$.
2. Antes de verificar la condición anterior, el efecto del ruido disminuye cuando la norma de $\mathbf{w}^{(k)}$ crece con respecto a $(\mathbf{n}_{kx} - \mathbf{n}_0)$. Es decir, cuanto más lejos estén los puntos del vértice $\mathbf{e}_0^{(c)}$ menor será el error introducido por el ruido. El algoritmo propuesto utiliza este concepto de forma que sólo se calcula la proximidad angular de aquellos vectores cuyas normas estén próximas a $\mathbf{v}_1 - \mathbf{e}_0^{(c)}$ o $\mathbf{v}_3 - \mathbf{e}_0^{(c)}$ (Figura 4.4).

Obviamente, el tiempo de convergencia es mayor en presencia del ruido, ya que el problema no se resuelve hasta detectar el vértice y los vectores arista corrompidos por la cota inferior de error (\mathbf{n}_m).

4.4 VECTORES CRÍTICOS

De las secciones anteriores se puede deducir que para identificar el medio y llevar a cabo la separación es necesario determinar:

1. Cualquier vértice, v_{ck} , del hiperparalelepípedo para poder realizar el cambio de coordenadas.
2. Cualesquiera p vectores, cada uno de ellos de una de las p aristas que converjan en el vértice de traslación.

Estas condiciones implican que tienen que generarse ciertas combinaciones de las señales originales, de forma que sin ellas no se podría identificar el medio. Como los vértices del espacio de observaciones se corresponden con los vértices del espacio de las fuentes (Propiedad 2.3, Sec.2.2), para obtener un vértice del hiperparalelepípedo de observación, las fuentes tienen que generar un vector tal que todas sus componentes tengan valores extremos, que se correspondan con sus cotas. El conjunto formado por estos vectores más el vértice se denominará de aquí en adelante conjunto de *vectores críticos*. Las condiciones para obtener los vectores arista son menos restrictivas, ya que es suficiente encontrar un solo vector en cada una de las aristas. Para p fuentes, habrá $p+1$ vectores críticos. Nótese que puede haber dependencias estadísticas entre las fuentes, que impidan la generación de los vectores críticos, en cuyo caso el procedimiento propuesto no es válido. Sin embargo, hay situaciones con dependencias estadísticas, en las que se satisfacen todos los requisitos y por tanto no se puede afirmar que el procedimiento no sea válido, en general, cuando hay dependencia estadística entre las fuentes. En la Sec.4.7 (Ejemplo 2) se incluye un caso de separación de 2 señales estadísticamente dependientes.

Conociendo las funciones de densidad de probabilidad, f_{s_i} , de las fuentes puede hacerse una estimación del tiempo medio necesario para obtener los vectores críticos.

Para simplificar la explicación considérese el siguiente caso. Supóngase que se tiene una mezcla de $p=3$ señales, estadísticamente independientes, uniformemente distribuidas en el

intervalo $[-5,+5]$, que se procesan con una precisión de $\Delta s=0.04$ (obtenible con un conversor A/D de 8 bits) y que las señales observadas se muestrean a una frecuencia de $F_s=22 \text{ KHz}$ ($T_s=4.5 \times 10^{-5}$ segundos).

La función de densidad de probabilidad para cualquier valor de s_i valdrá (A1.22):

$$f_{s_i}(s_i) = \frac{1}{5 - (-5)} = \frac{1}{10} \quad (4.19)$$

Por otra parte, la probabilidad de que se genere un punto de una arista, $ar_1=(s_1, s_{2m}, s_{3m})$, por ejemplo, será:

$$\begin{aligned} p_{ar1} &= p[-5 \leq s_2 \leq -4.96] \cdot p[-5 \leq s_3 \leq -4.96] = \\ &= (f_{s_2}(-5) \cdot \Delta s_2) \cdot (f_{s_3}(-5) \cdot \Delta s_3) = 1.6 \times 10^{-5} \end{aligned} \quad (4.20)$$

donde se ha hecho uso de la independencia estadística de las señales (A1.22) y de la expresión (4.19).

La probabilidad de obtener un vértice, $v_{s0}=(s_{1m}, s_{2m}, s_{3m})$, por ejemplo, es:

$$\begin{aligned} p_{v_{s0}} &= p[-5 \leq s_1 \leq -4.96] \cdot p[-5 \leq s_2 \leq -4.96] \cdot p[-5 \leq s_3 \leq -4.96] = \\ &= (f_{s_1}(-5) \cdot \Delta s_1) \cdot (f_{s_2}(-5) \cdot \Delta s_2) \cdot (f_{s_3}(-5) \cdot \Delta s_3) \approx 6 \times 10^{-8} \end{aligned} \quad (4.21)$$

Las expresiones (4.20) y (4.21) indican que, por término medio, cada 62500 y 17×10^6 muestras se obtendrá un punto de una arista y un vértice respectivamente. Teniendo en cuenta que cada muestra se capta cada $T_s=4.5 \times 10^{-5}$ segundos, se tiene que el tiempo medio para la obtención de una arista es de $t_{ar1}=62500 \times 4.5 \times 10^{-5}=2.8$ segundos, y para la obtención de un vértice $t_{v_{s0}}=17 \times 10^6 \times 4.5 \times 10^{-5}=765$ segundos.

Se concluye que el método propuesto en este capítulo convergerá más rápidamente cuanto mayores sean los valores de $f_{s_i}(s_{im})$ y $f_{s_i}(s_{iM})$; o en otras palabras, cuanto más "suprauniformes" sean las fuentes (Sec.A1.3.1).

4.4.1 Caso de tres fuentes, $p = 3$.

En el caso de tres fuentes, para poder obtener los vectores críticos tiene que darse una de las siguientes combinaciones, que corresponden a las coordenadas de los 8 vértices del hiperparalelepípedo:

$$\begin{aligned} v_{S0} &= (s_{1m}, s_{2m}, s_{3m}) & v_{S4} &= (s_{1M}, s_{2m}, s_{3m}) \\ v_{S1} &= (s_{1m}, s_{2m}, s_{3M}) & v_{S5} &= (s_{1M}, s_{2m}, s_{3M}) \\ v_{S2} &= (s_{1m}, s_{2M}, s_{3m}) & v_{S6} &= (s_{1M}, s_{2M}, s_{3m}) \\ v_{S3} &= (s_{1m}, s_{2M}, s_{3M}) & v_{S7} &= (s_{1M}, s_{2M}, s_{3M}) \end{aligned} \quad (4.22)$$

Como se dijo en el apartado anterior, además de encontrar uno de los ocho vectores, es suficiente encontrar un solo vector en cada una de las p aristas. Entonces, si la traslación se realiza, por ejemplo, con el vértice v_{S4} sólo haría falta generar tres vectores de la forma (s_1, s_{2m}, s_{3m}) , (s_{1M}, s_2, s_{3m}) y (s_{1M}, s_{2m}, s_3) , donde s_i puede tomar cualquier valor, $s_i \in [s_{im}, s_{iM}]$. En este caso el conjunto de vectores críticos estaría formado por el vértice v_{S4} y los tres vectores anteriores.

4.5 IMPLEMENTACIÓN DEL ALGORITMO CON REDES NEURONALES.

El procedimiento de estimación de las fuentes consta de dos algoritmos que se pueden identificar con dos redes neuronales trabajando en paralelo. Una realiza la separación de las señales de forma recursiva para obtener las señales transformadas, $y(t)$. La otra red va calculando adaptativamente la matriz de pesos W , bajo aprendizaje no supervisado, e identifica el medio lineal. A medida que se van obteniendo las sucesivas matrices W , éstas se van aproximando gradualmente a una matriz similar a A . Los elementos de la matriz W representan los pesos que son comunes a las dos redes (Figura 4.5).

La red de aprendizaje se encarga de ir obteniendo los vectores peso normalizados, $w_j = (w_{ij})$. Esto no presenta ningún problema ya que los elementos de cada columna de la matriz W están divididos por un valor constante $|w_j|$ y por tanto, la relación entre A y W dada por la siguiente expresión:

$$W^{-1} \cdot A = P \quad (4.23)$$

se sigue manteniendo, es decir, W sigue siendo similar a A (Propiedad 2.4, Sec.2.2).

A continuación se va a describir el comportamiento de ambas redes de forma separada, recordando que ambas actúan en paralelo, en la forma que se esquematiza en la Figura 4.5.

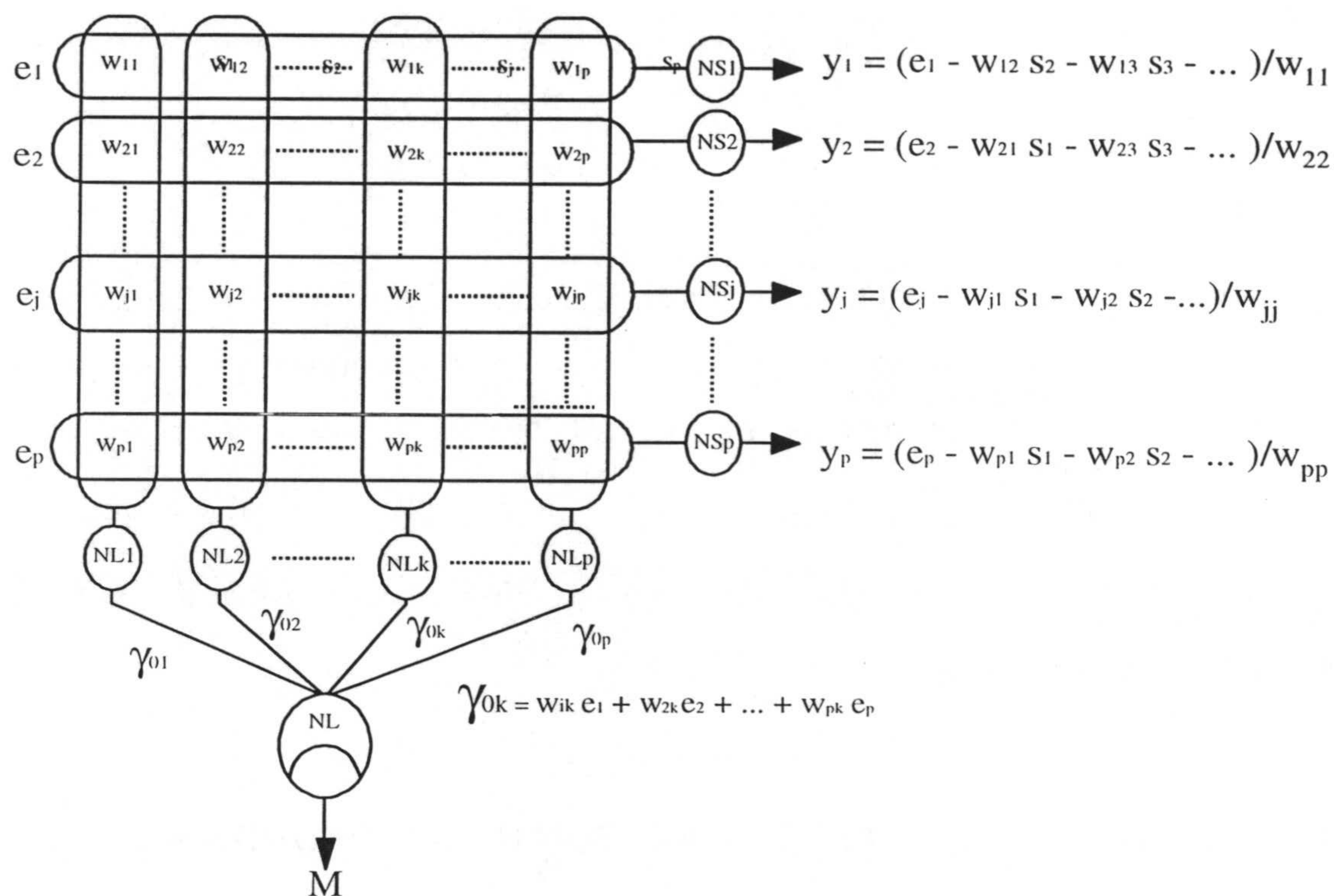


Figura 4.5. Paralelismo entre la obtención de las fuentes y los pesos.

4.5.1 Red de separación de las señales.

Como se vio en la Sec.2.1, las observaciones, $e(t)$, se pueden expresar en función de las señales originales, $s(t)$, según (2.2). Dicha expresión se puede reescribir de la siguiente forma:

$$e_i(t) = \sum_{j=1}^p w_{ij}(t) \cdot y_j(t) = w_{ii} \cdot y_i(t) + \sum_{j=1, j \neq i}^p w_{ij}(t) \cdot y_j(t) \quad \forall i \in \{1, \dots, p\} \quad (4.24)$$

a partir de esta expresión se puede calcular de forma recursiva una de las fuentes, $y_i(t)$, conocidas el resto, $y_j(t)$, (3.9):

$$y_i(t) = \frac{1}{w_{ii}(t)} \cdot e_i(t) - \sum_{j=1, j \neq i}^p \frac{w_{ij}(t)}{w_{ii}(t)} \cdot y_j(t) \quad \forall i \in \{1, \dots, p\} \quad (4.25)$$

La expresión (4.25) se puede representar con la red neuronal de la Figura 4.6, formada por p neuronas separadoras ($N_{S1}, N_{S2}, \dots, N_{Sp}$). Las entradas, $e_i(t)$, son las señales observadas, y las salidas, $w_{ii} \cdot y_i$, son las señales reconstruidas multiplicadas por una constante.

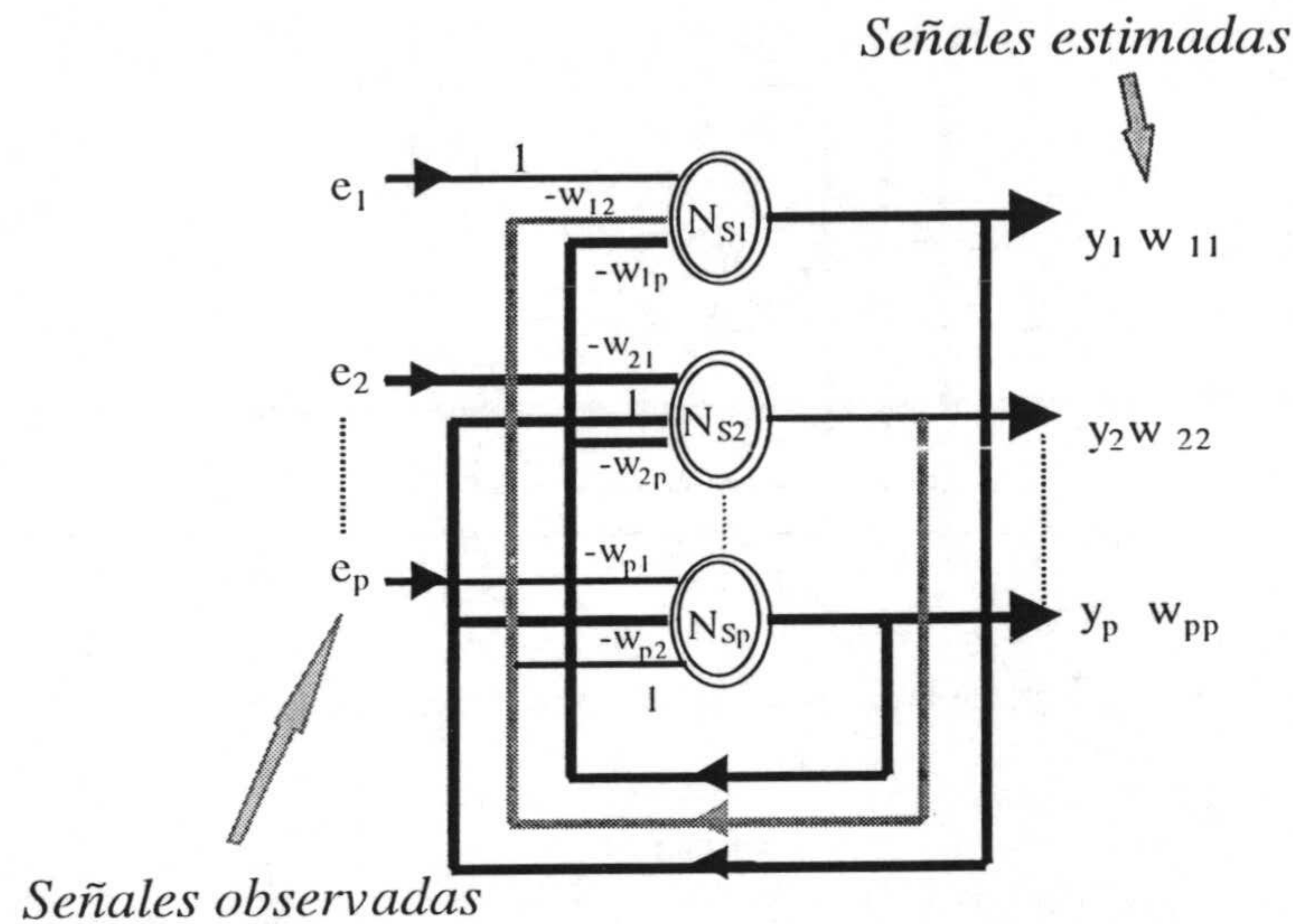


Figura 4.6. Red neuronal de estimación de las fuentes.

Nótese, que con esta red recursiva, no es necesario calcular explícitamente W^{-1} para obtener las fuentes y que, para un funcionamiento adecuado es necesario que se verifique $w_{ii} \neq 0$, $i=1, \dots, p$, condición acorde con la Hipótesis 2.5 (Sec.2.1.1).

4.5.1.1 Caso $p=3$

En el caso de tres fuentes, la expresión (4.24) tendría la siguiente forma:

$$\begin{aligned} y_1(t) &= \frac{e_1(t)}{w_{11}} - \frac{w_{12}}{w_{11}} \cdot y_2(t) - \frac{w_{13}}{w_{11}} \cdot y_3(t) \\ y_2(t) &= \frac{e_2(t)}{w_{22}} - \frac{w_{21}}{w_{22}} \cdot y_1(t) - \frac{w_{23}}{w_{22}} \cdot y_3(t) \\ y_3(t) &= \frac{e_3(t)}{w_{33}} - \frac{w_{31}}{w_{33}} \cdot y_1(t) - \frac{w_{32}}{w_{33}} \cdot y_2(t) \end{aligned} \quad (4.26)$$

La red neuronal en este caso tomaría la forma de la Figura 4.7, en la que hay tres neuronas separadoras (N_{S1} , N_{S2} y N_{S3}), tres componente de observación de entrada $e=(e_1, e_2$ y $e_3)$ y tres componentes de salida $y=(y_1, y_2$ e $y_3)$ que se van obteniendo según la ecuación (4.25).

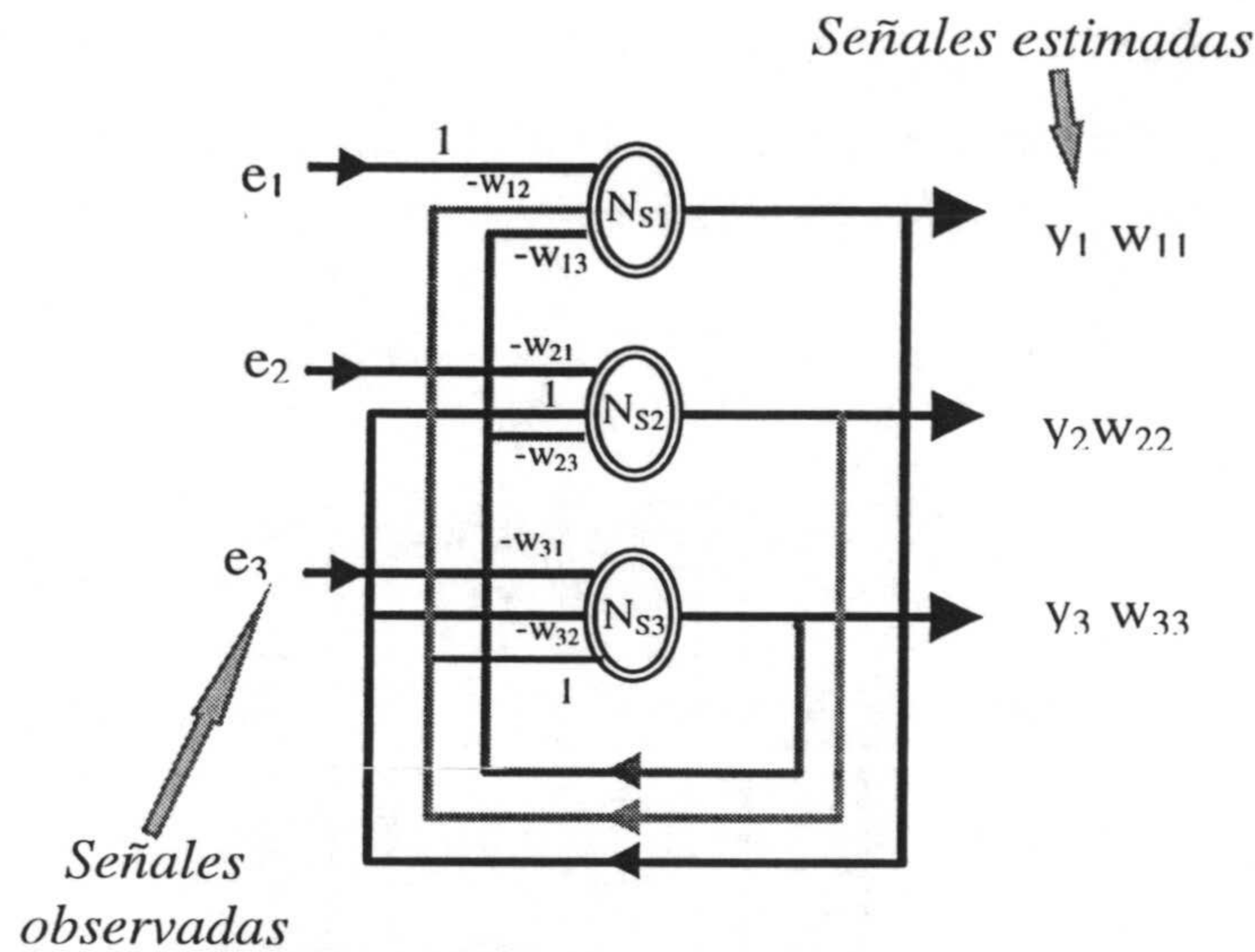


Figura 4.7. Red de separación para $p=3$.

4.5.2 Red de aprendizaje.

Para facilitar la obtención de los pesos, el algoritmo primero normaliza los vectores de entrada, de forma que $\sum_{i=1}^p e_i^{*2} = 1$. Para llevar esto a cabo se hace la siguiente transformación a la entrada de la red:

$$e_i^*(t) \leftarrow \frac{e_i(t)}{|e(t)|} \quad \forall i \in \{1, \dots, p\} \quad (4.27)$$

Cuando se ha trasladado y normalizado el vector de entrada, se lleva a cabo la obtención de los pesos, de forma adaptativa y no supervisada.

En la Figura 4.8 se muestra la red neuronal de aprendizaje. Consta de p entradas que son las señales observadas normalizadas. Las dos siguientes capas son de neuronas. Una de ellas está formada por p neuronas, N_{L1}, \dots, N_{Lp} , lineales, y la otra (la tercera capa, de

salida) contiene un sólo elemento de cómputo, N_L , no lineal, de tipo escalón y caracterizado por una matriz umbral Γ , y una salida M , que indica si hay ($M=1$) o no ($M=0$) que modificar los pesos.

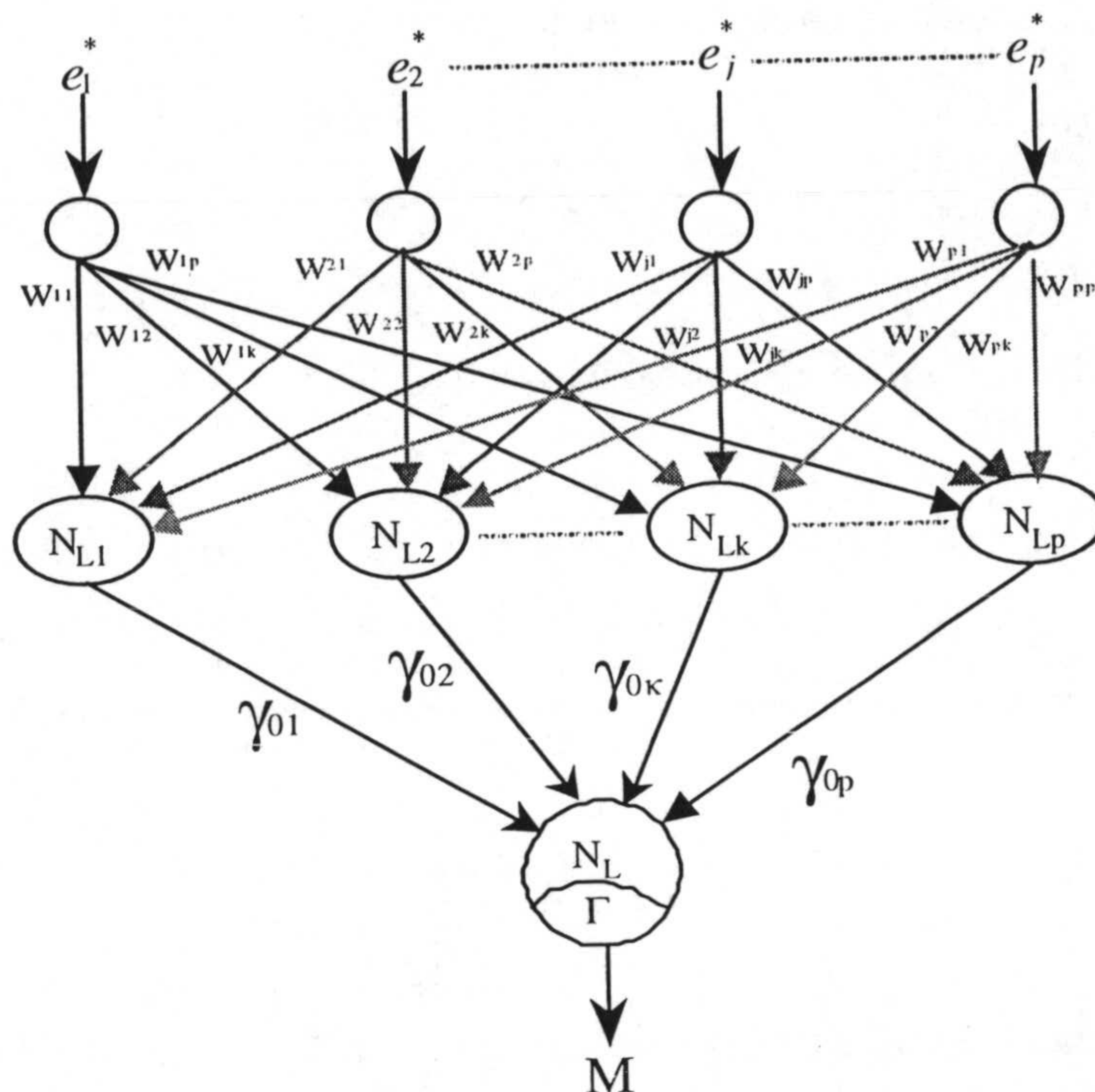


Figura 4.8. Red neuronal de aprendizaje de pesos.

Las salidas ($\gamma_{01}, \dots, \gamma_{0k}, \dots, \gamma_{0p}$) de las neuronas de aprendizaje (N_{L1}, \dots, N_{Lp}) de la segunda capa son la suma ponderada de las entradas, es decir:

$$\begin{aligned} \gamma_{01} &= \sum_{i=1}^p w_{1i} \cdot e_i^* \\ &\dots\dots\dots \\ \gamma_{0k} &= \sum_{i=1}^p w_{1k} \cdot e_i^* \\ &\dots\dots\dots \\ \gamma_{0p} &= \sum_{i=1}^p w_{1p} \cdot e_i^* \end{aligned} \tag{4.28}$$

Por tanto, cada salida, γ_{0k} , representa el producto escalar del vector de entrada normalizado, e^* , con el vector w_k . Es decir, γ_{0k} es la proximidad angular entre el nuevo vector de observación y la arista k del cono del hiperparalelepípedo (4.1).

La neurona N_L se encarga de generar una señal M para poder modificar los pesos. Esta señal está caracterizada por una *matriz umbral*, Γ . Los elementos de esta matriz son las proximidades entre los vectores arista, es decir:

$$\Gamma_w = (\gamma_{jk}) \quad j, k \in \{1, \dots, p\} \quad (4.29)$$

donde γ_{jk} es la proximidad entre los vectores w_j y w_k .

Se puede definir un umbral asociado a cada vector w_k basado en la matriz anterior, cuya proximidad respecto al resto de los vectores w_j se podría representar de la siguiente forma:

$$\Gamma_w(k) = \sum_{i=1, i \neq k}^p \gamma_{ki} \quad (4.30)$$

El elemento de salida N_L realiza tres funciones:

1. Detecta la neurona ganadora, N_k , de la segunda capa. Es decir, detecta la neurona k que tenga la mayor salida, γ_{0k} .
2. Suma sus entradas (las salidas de las neuronas del segundo nivel) excepto la de la ganadora:

$$\Gamma_{e \leftarrow k} = \gamma_{01} + \dots + \gamma_{0k-1} + \gamma_{0k+1} + \dots + \gamma_{0p} \quad (4.31)$$

3. Si la suma cae por debajo del umbral Γ_k , se actualiza la matriz umbral y se genera el valor de salida $M=1$, ya que hay que modificar los pesos:

$$\text{iff } \Gamma_{e \leftarrow k} < \Gamma_k : \gamma_{ki1} \leftarrow \gamma_{0i}, \gamma_{ik} \leftarrow \gamma_{ki}, \quad M \leftarrow 1 \quad (4.32)$$

La modificación de los pesos se lleva a cabo según:

$$\text{iff } M = 1 : w_{1k} \leftarrow e_1^*, \dots, w_{jk} \leftarrow e_j^*, \dots, w_{pk} \leftarrow e_p^* \quad (4.33)$$

Como el vector e^* está normalizado, los vectores w_k también lo están.

Resumiendo, la red sustituye el vector arista que está más cerca del último vector de entrada si con ello se obtiene un nuevo conjunto de vectores arista con proximidad menor.

Nótese, que de acuerdo con la topología mostrada en la Figura 4.8, cada neurona sólo realiza operaciones locales. En la capa oculta cada neurona (N_{Lk}) sólo lleva a cabo la suma ponderada de sus entradas, mientras que la neurona de la capa de salida determina el máximo de sus entradas, las suma y las compara con los valores umbrales. Estos valores se modifican teniendo en cuenta solamente los datos de entrada de esta neurona. La señal de salida de la red ($M \in \{1, 0\}$) actúa sobre todas las conexiones entre las capas 1 y 2, indicando si hay que cambiar o no los pesos correspondientes al vector ganador por las señales de entrada que pasan localmente por estas conexiones.

La visión geométrica descrita, recuerda la forma de interpretar geoméricamente el aprendizaje competitivo [RUM86]. En la red descrita, como en el aprendizaje competitivo, cada vector peso representa una clase, y siempre que llega un nuevo vector estímulo, $e(t)$, el vector peso que esté más cerca del vector estímulo en la hiperesfera gana la competición. En el aprendizaje competitivo, siempre que una unidad gana, su vector peso se *mueve* hacia el estímulo actual, mientras que en el procedimiento propuesto aquí si la nueva distancia angular entre los pesos aumenta, el vector peso es *sustituido* por el vector estímulo.

4.5.2.1 Caso $p=3$

Como se ha visto en el caso general, una vez trasladados y normalizados los vectores, se van calculando los pesos con el proceso representado en la red de la Figura 4.9. Las salidas de la primera capa de neuronas son las proximidades angulares entre los vectores de observación y los pesos, que en este caso serán las tres siguientes (4.1):

$$\begin{aligned}\gamma_{01} &= \mathbf{w}_1 \cdot \mathbf{e}^* = w_{11} \cdot e_1^* + w_{21} \cdot e_2^* + w_{31} \cdot e_3^* \\ \gamma_{02} &= \mathbf{w}_2 \cdot \mathbf{e}^* = w_{12} \cdot e_1^* + w_{22} \cdot e_2^* + w_{32} \cdot e_3^* \\ \gamma_{03} &= \mathbf{w}_3 \cdot \mathbf{e}^* = w_{13} \cdot e_1^* + w_{23} \cdot e_2^* + w_{33} \cdot e_3^*\end{aligned}\tag{4.34}$$

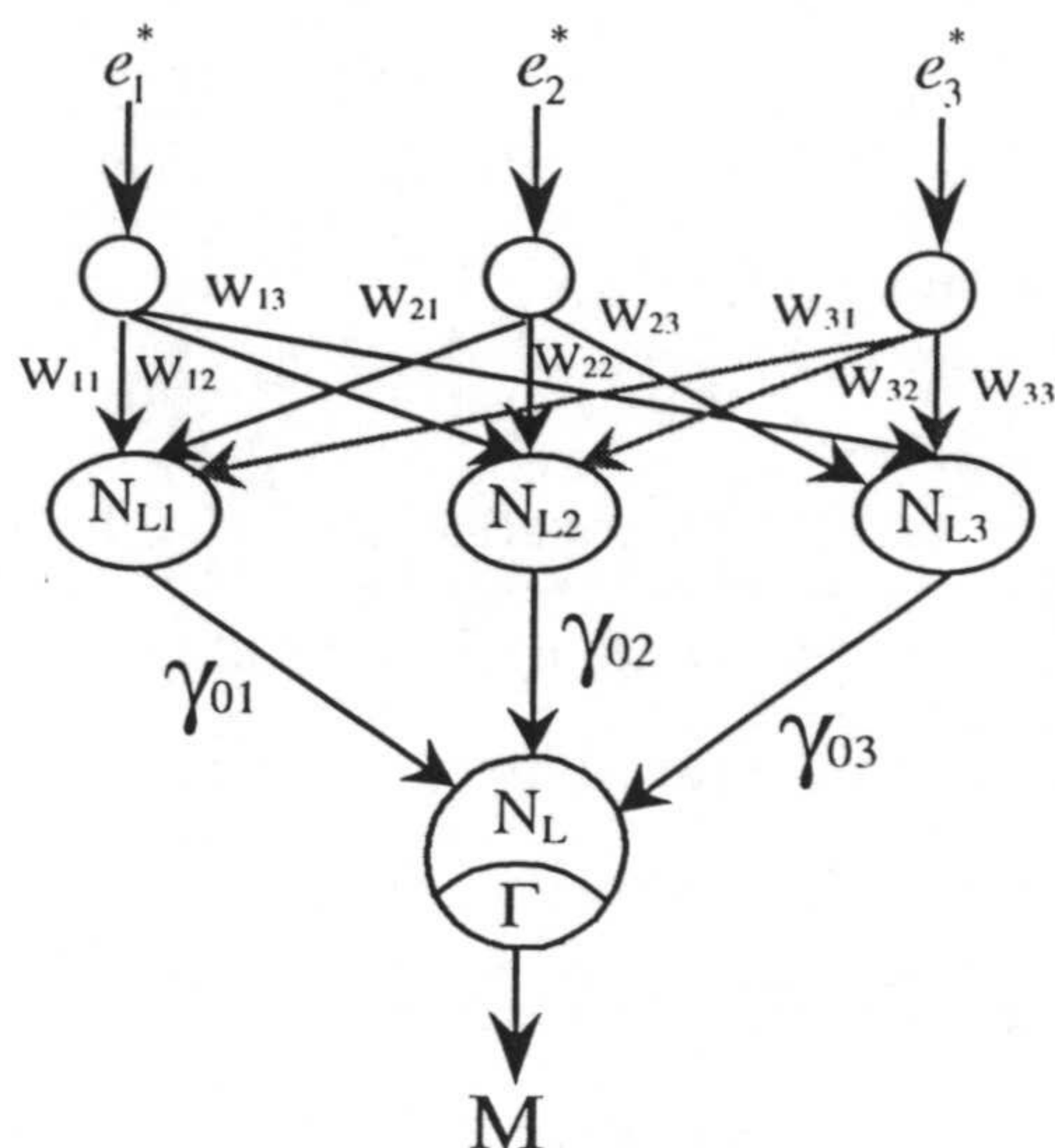


Figura 4.9. Red neuronal de aprendizaje para $p=3$.

Una vez obtenidas las proximidades angulares sus valores pasan a la siguiente capa, donde la neurona N_L generará la señal de salida M según sea la matriz umbral, Γ .

$$\begin{aligned}
 \Gamma_1 &= \gamma_{12} + \gamma_{13} \\
 \Gamma_2 &= \gamma_{21} + \gamma_{23} \\
 \Gamma_3 &= \gamma_{31} + \gamma_{32}
 \end{aligned}
 \tag{4.35}$$

Según (4.35), Γ_1 será la proximidad del vector w_1 respecto w_2 y w_3 , Γ_2 la proximidad del vector w_2 respecto w_1 y w_3 y Γ_3 la proximidad del vector w_3 respecto w_1 y w_2 . Cuando llega una nueva entrada, e , la red determina el vector peso más próximo (w_3 en la Figura 4.10a, y w_2 en la Figura 4.10b) y si la proximidad es menor que la que había, e sustituye al vector peso más próximo. En la Figura 4.10a, se obtiene una proximidad menor con el vector e , es decir, $\gamma_{01} + \gamma_{02} < \Gamma_3$, donde γ_{01} y γ_{02} son las proximidades del vector e respecto a w_1 y w_2 , y por tanto, e sustituye a w_3 . Sin embargo, esto no ocurre en la Figura 4.10b, ya que al sustituir w_2 por e no se obtiene una proximidad inferior, $\gamma_{01} + \gamma_{03} > \Gamma_2$, donde γ_{01} y γ_{03} son las proximidades del vector e respecto a w_1 y w_3 , y por tanto el vector w_2 no se actualiza.

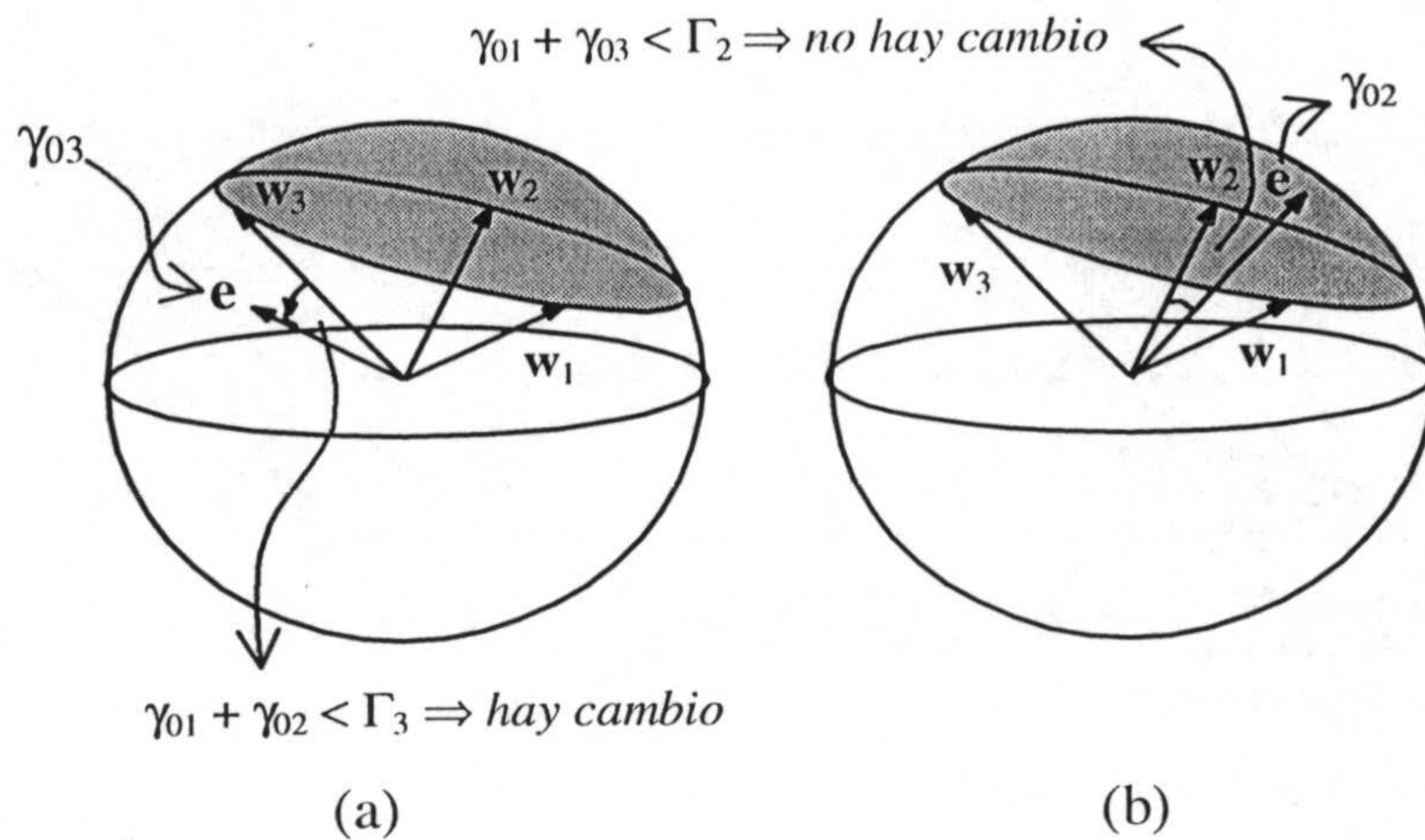


Figura 4.10. *Modificación de los pesos.*

4.6 IMPLEMENTACIÓN

Para probar este procedimiento se ha utilizado el entorno de simulación descrito en el Capítulo 3. La separación de señales, en el presente caso, se efectúa en las fases que se indican a continuación.

Preproceso.

En el preproceso, además de la obtención adaptativa del vértice y la normalización de los vectores de mezcla, se trata de reducir los efectos de la localización inexacta del vértice y del ruido que pueda haber en la mezcla. Como se muestra en la Figura 4.11a, donde se ven dos aristas (i, j) en el plano que las contiene, la detección de un vértice distinto al teórico (correcto) implica que las coordenadas de las aristas sean erróneas (y consecuentemente que también lo sean los elementos de la columna correspondiente de la matriz W). Se puede demostrar fácilmente que cuanto mayor sea el módulo del vector detectado como vector peso, menor será el error.

Lo mismo sucede con el ruido de la mezcla, como se demostró en la Sec.4.3. Suponiendo que el ruido es aditivo y acotado, el vector corrompido por el ruido $e^{(i)(c)}$ se desplazará dentro de un intervalo, como se puede ver en la Figura 4.11b. Obviamente, el ruido también afectará al vértice v_0 de la Figura 4.11, desplazándolo de su posición original a la posición representada por $v_0^{(c)}$.

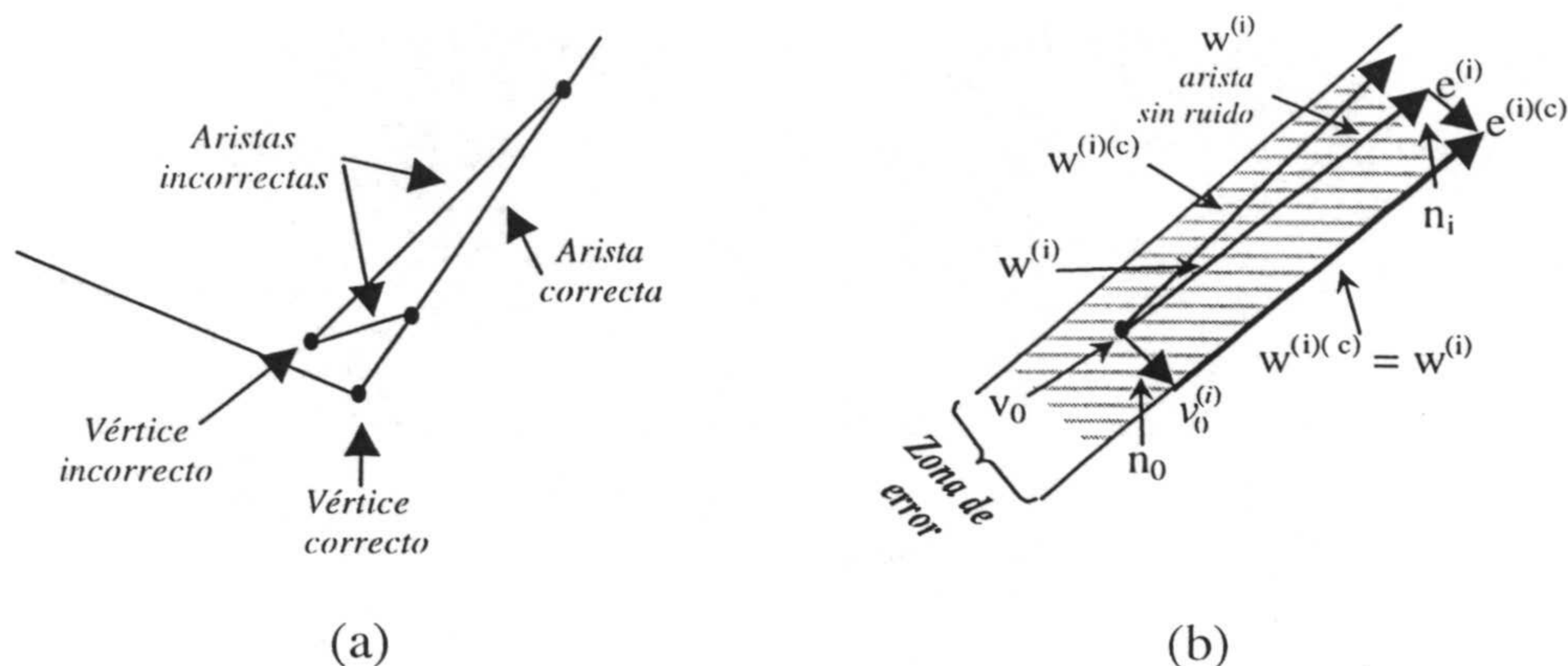


Figura 4.11. Detección del vértice correcto.

Como se indicó en la Sec.4.3 (4.18), si el ruido es acotado y uniforme (es decir, afecta de igual forma a todos los puntos del medio) y se detectan como valores extremos el vértice y el vector arista, el ruido tiende a ser compensado $n_i \approx n_0$. En este caso, según (4.18) el vector obtenido sería correcto, $w^{(i)(c)} \approx w^{(i)}$. Sin embargo, se necesita más tiempo para localizar los valores extremos en presencia de ruido.

Se puede deducir de (4.18) que el efecto del ruido disminuye cuando la norma de $w^{(i)}$ con respecto a $n_i - n_0$ aumenta. En la implementación, además del vértice de cambio de coordenadas, v_0 se obtiene también el vértice opuesto, v_1 . Para considerar un vector de observación como un candidato a vector arista, debe tener una norma del orden de:

$$n_r = \frac{|v_1 - v_0|}{2} \quad (4.36)$$

Como los vértices se van obteniendo adaptativamente, al principio del proceso los puntos están muy juntos. Para compensar este efecto sólo se consideran aquellos vectores de observación, $e(t)$, cuyas normas sean mayores que el siguiente valor:

$$n_{test} = (n_0 - n_r) \cdot e^{-\frac{t}{\tau}} + n_r \quad (4.37)$$

donde n_0 y τ son dos parámetros que representan la norma inicial del test y una constante de tiempo, respectivamente.

Aprendizaje.

En esta fase se implementa la red de la Sección 4.5.2. Es decir, se realizan las siguientes operaciones:

- Normalización de los vectores de mezcla según (4.27)
- Obtención y actualización de los pesos a partir del cálculo de las proximidades angulares (4.28), (4.32).

Separación.

Para realizar la separación se aplica la ecuación (4.25) para cada valor de t . La indeterminación debida al índice de permutación impide que se pueda aplicar de forma correcta la expresión (4.25), ya que en cada iteración hay que asociar cada e_i con sus correspondientes coeficientes w_{ii} y e_{ij} ($j=1, \dots, p$). Para solucionar este problema, se define siempre la matriz de mezcla A de forma que $|a_{ii}| > |a_{ij}|$ para todo $i, j \in \{1, \dots, p\}$ (Sec.2.5.2.1). Así, el simulador en el momento en que obtiene las señales $y(t)$, reordena la matriz $W(t)$ de forma que el elemento con mayor valor absoluto de cada columna tiene que estar en la diagonal principal. Nótese que esta restricción no se aplica para obtener la matriz de separación, W , sino para recuperar las fuentes y poder eliminar una de las indeterminaciones que se produce cuando se intenta resolver el problema de separación (Sec.1.2). Además, la condición que se ha impuesto, indica físicamente que cada sensor i está más cerca o es más sensible a la fuente i (coeficientes a_{ii}) que al resto ($j \neq i$).

Cuando se aplica iterativamente la expresión (4.25), se toma como valor inicial para cada valor de t : $y_0(t) = e(t)$. El proceso termina cuando la diferencia entre los valores de y de dos iteraciones consecutivas es menor que un parámetro de convergencia ε . Obviamente, cuanto menor sea el parámetro de convergencia más iteraciones se necesitarán para conseguir un valor aceptable.

4.7. RESULTADOS EXPERIMENTALES.

El primer ejemplo (Ejemplo 1, Figura 4.11) muestra los resultados de la separación de 5 señales sintéticas: seno, triangular, y las tres siguientes propuestas por Amari, Cichocki y Yang [AMA95] (Sec.3.3.1):

$$\begin{aligned}
s_3 &= n(t) \\
s_4 &= 5 \sin\left(\frac{2\pi}{100}t\right) - \cos\left(\frac{2\pi}{800}t\right) \\
s_5 &= 10 \operatorname{sign}\left[\sin\left(\frac{2\pi}{40}t\right) + 9 \cos\left(\frac{2\pi}{400}t\right)\right]
\end{aligned} \tag{4.38}$$

donde $n(t)$ es una fuente de ruido aleatorio distribuida uniformemente en el rango $[0, 10]$.

La matriz original A es la siguiente:

$$A = \begin{pmatrix} 1.00 & 0.48 & 0.52 & 0.38 & 0.42 \\ 0.47 & 0.90 & 0.53 & 0.45 & 0.39 \\ 0.46 & 0.51 & 0.85 & 0.39 & 0.43 \\ 0.47 & 0.52 & 0.47 & 0.95 & 0.38 \\ 0.38 & 0.46 & 0.54 & 0.50 & 0.93 \end{pmatrix} \tag{4.39}$$

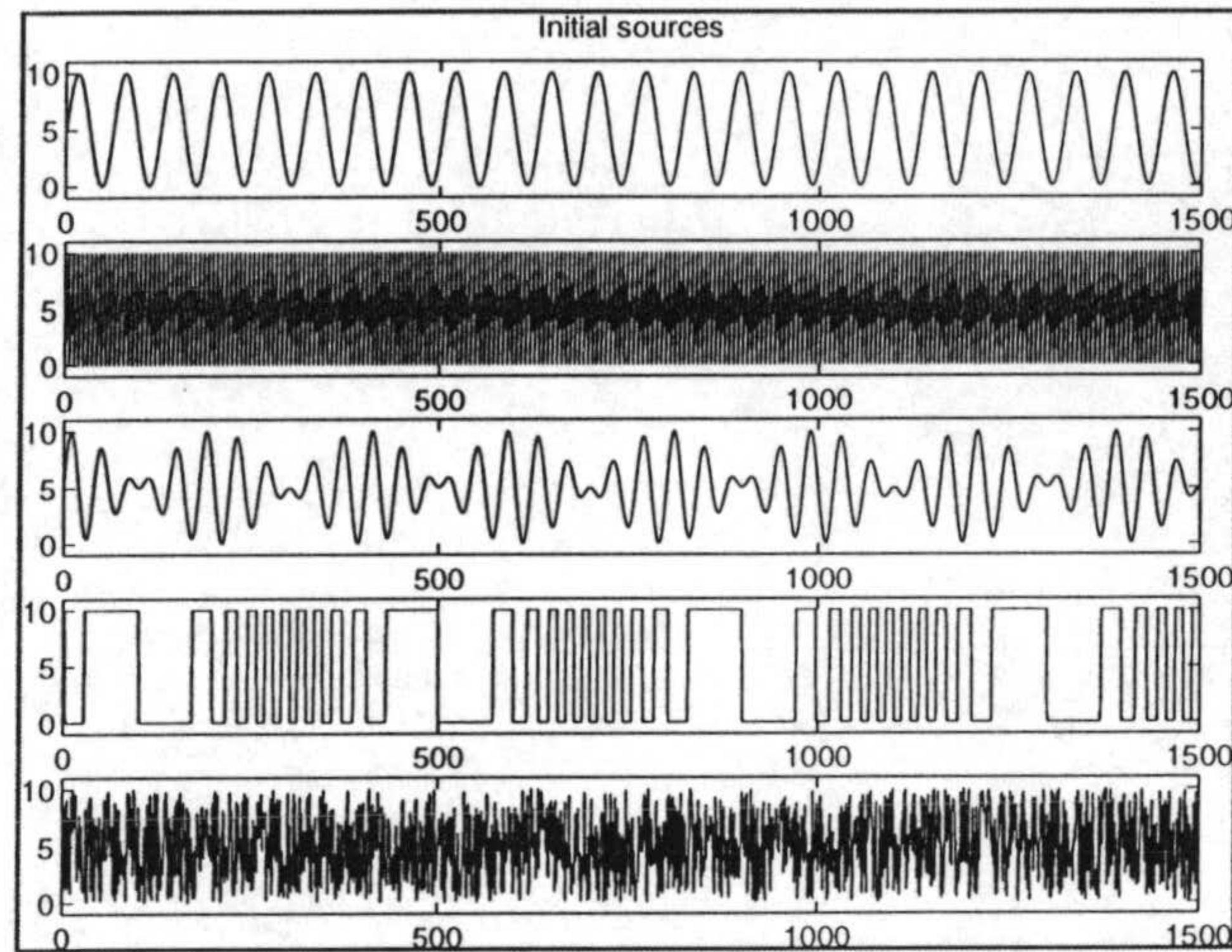
Si se normaliza la matriz A por columnas se tiene:

$$AN = \begin{pmatrix} 0.75 & 0.36 & 0.39 & 0.30 & 0.34 \\ 0.35 & 0.67 & 0.40 & 0.35 & 0.32 \\ 0.34 & 0.38 & 0.64 & 0.30 & 0.35 \\ 0.35 & 0.39 & 0.35 & 0.74 & 0.31 \\ 0.28 & 0.34 & 0.40 & 0.39 & 0.75 \end{pmatrix} \tag{4.40}$$

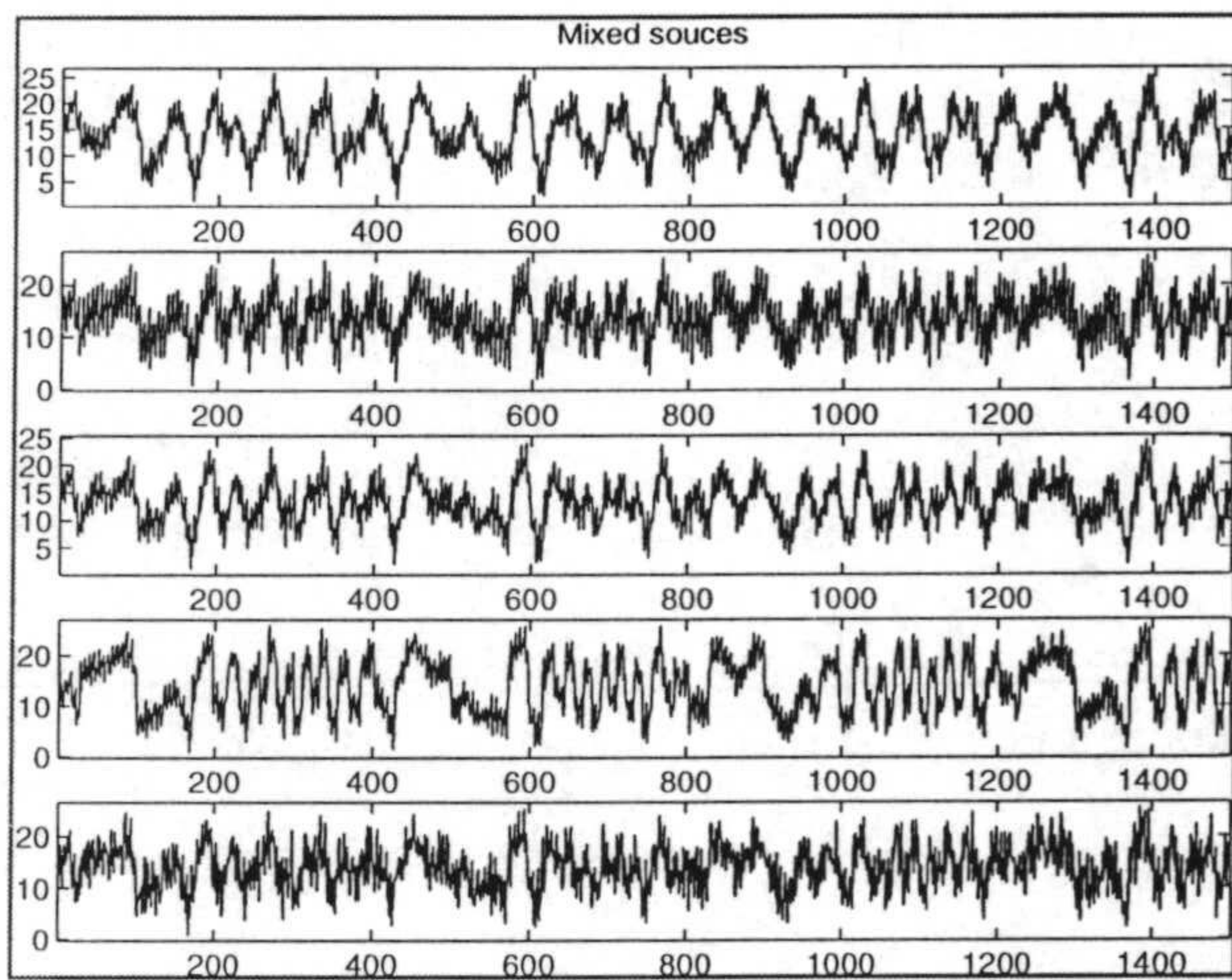
Tras reordenar, se obtiene la siguiente matriz W :

$$W = \begin{pmatrix} 0.72 & 0.37 & 0.39 & 0.31 & 0.36 \\ 0.36 & 0.66 & 0.40 & 0.36 & 0.32 \\ 0.36 & 0.38 & 0.63 & 0.32 & 0.35 \\ 0.35 & 0.39 & 0.35 & 0.71 & 0.31 \\ 0.31 & 0.36 & 0.42 & 0.41 & 0.74 \end{pmatrix} \tag{4.41}$$

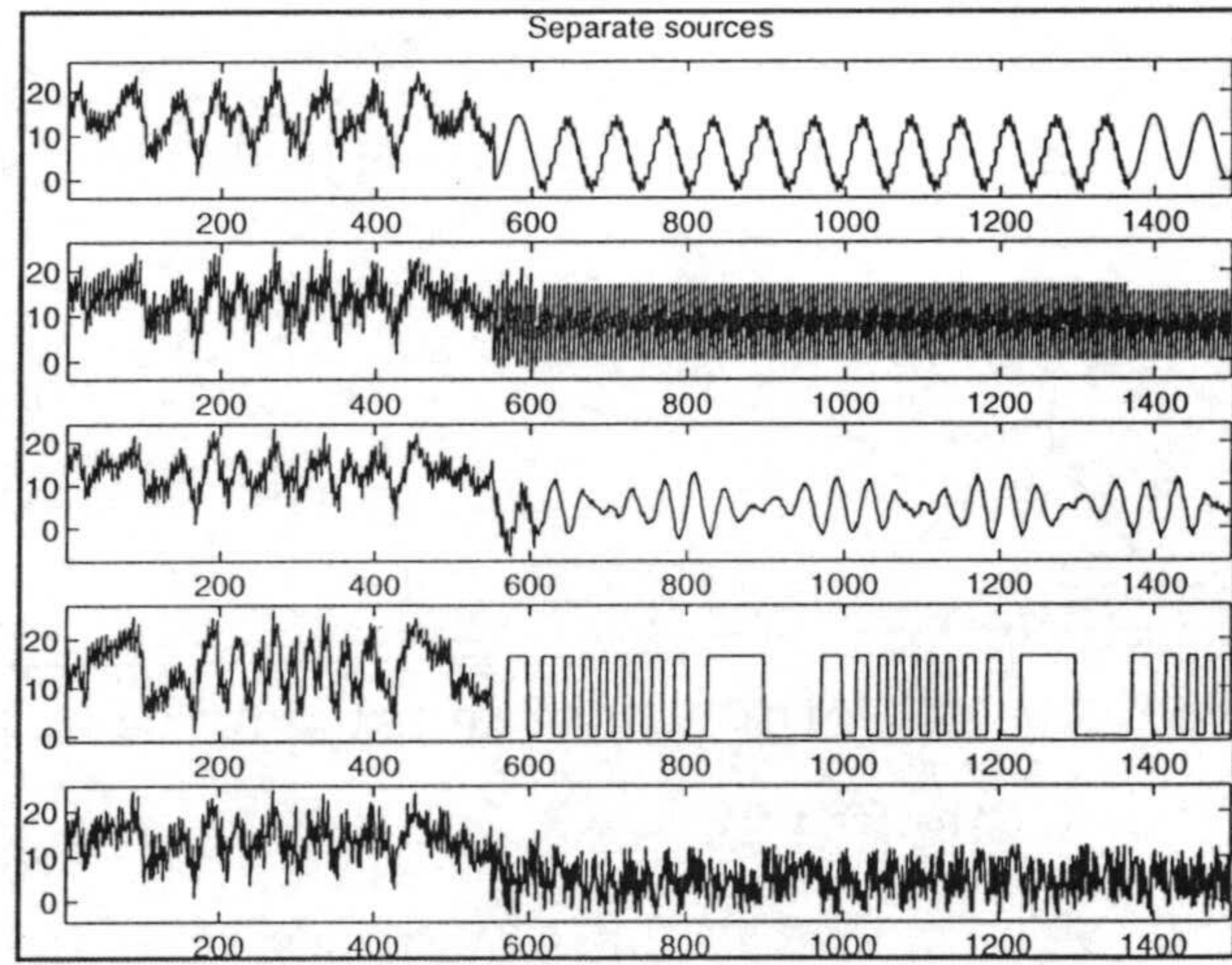
En la Figura 4.12 se pueden ver las representaciones de las fuentes originales (Figura 4.12a), las mezclas (Figura 4.12b), y las señales recuperadas (Figura 4.12c).



(a)



(a)



(c)

Figura 4.12. (a) fuentes originales, (b) las mezclas, (c) y las señales recuperadas.

En la Figura 4.13 se puede ver la variación con el tiempo del índice de prestaciones (PIA, Sec.2.5.1 y 3.7). Como se vio, este índice proporciona una medida de la calidad de la separación de forma que cuanto más próximo a cero sea mejor es la separación. En este ejemplo, el índice llega a ser $PIA(NM=19.000)=0.5521$.

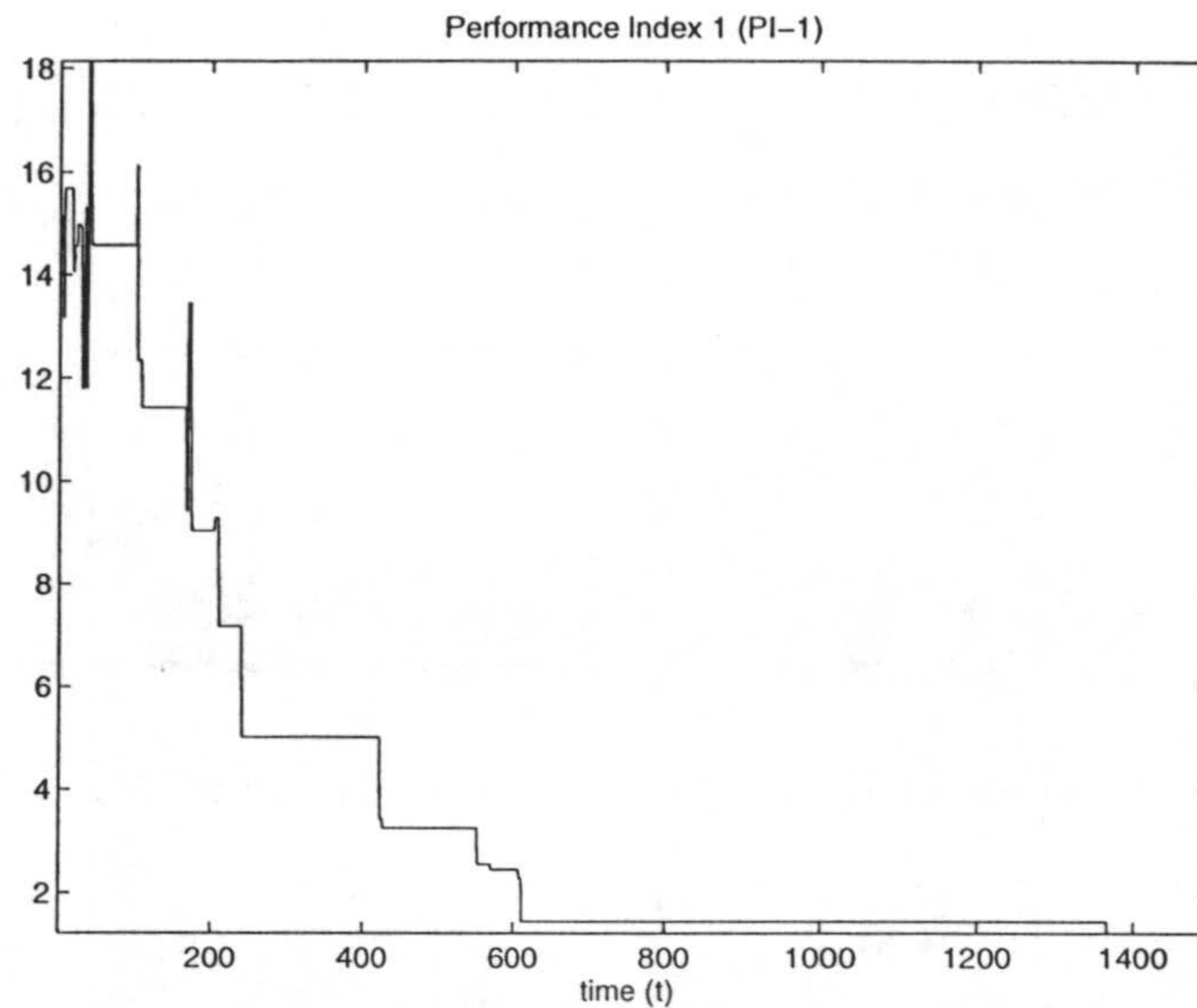


Figura 4.13. Índice de prestaciones de Amari (PIA).

El Ejemplo 2 (Figura 4.14) trata de la separación de dos señales estadísticamente dependientes, cuyos vectores de entrada corresponden a una circunferencia:

$$\begin{aligned} s_1(t) &= 5 \cdot \cos(2\pi n(t)) \\ s_2(t) &= 5 \cdot \sin(2\pi n(t)) \end{aligned} \quad (4.42)$$

donde $n(t)$ está formado por un conjunto de números aleatorios uniformemente distribuidos en el intervalo $[0,1]$.

La matriz de mezcla A , normalizada por columnas es:

$$AN = \begin{pmatrix} -0.8077 & 0.3304 \\ -0.5896 & 0.9439 \end{pmatrix} \quad (4.43)$$

El índice de prestaciones toma el valor $PIA(t=499) = 0.00023$, dando lugar a la siguiente matriz de pesos:

$$W = \begin{pmatrix} 0.3303 & -0.8077 \\ 0.9439 & -0.5896 \end{pmatrix} \quad (4.44)$$

Nótese que las columnas de W obtenidas por la red están en un orden diferente al de la matriz de mezcla original A .

La Figura 4.14 incluye las representaciones de las fuentes originales (Figura 4.14a), el espacio de vectores fuente (Figura 4.14b), el espacio de mezclas (Figura 4.14c) y el espacio de las señales recuperadas (Figura 4.14d).

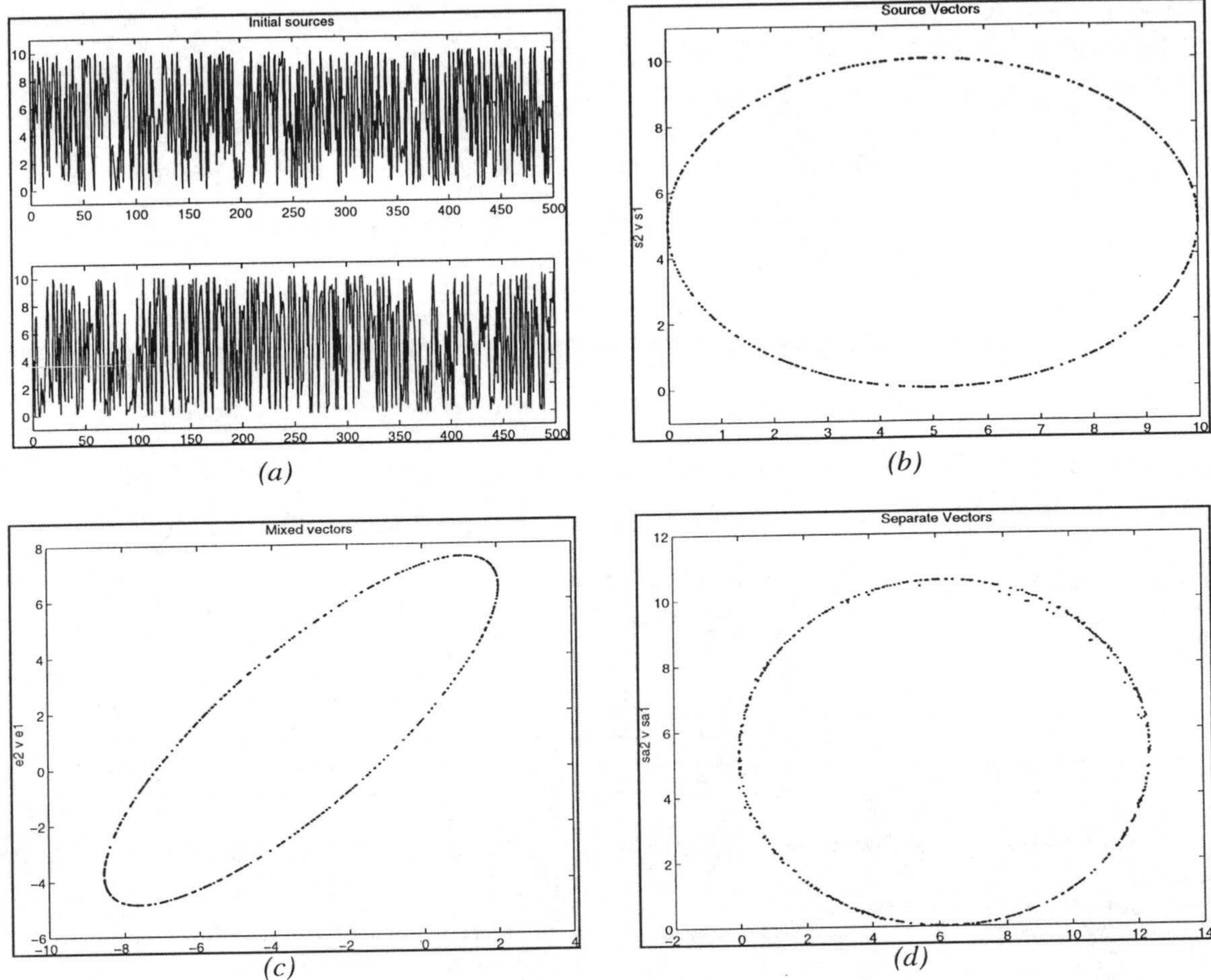


Figura 4.14. (a) fuentes originales, (b) espacio de las fuentes, (c) espacio de observaciones, (d) espacio de las señales recuperadas.

Queda probado con este ejemplo que el método propuesto es capaz de separar señales estadísticamente dependientes.

El Ejemplo 3 corresponde a tres señales reales, las palabras “cuerpo,” “mano”, y “muñeca”. Estas fuentes se han captado con un conversor A/D de 14 bits a una frecuencia de $f_s=8$ KHz, y una relación señal ruido de 24 dB. La matriz de mezcla considerada es:

$$AN = \begin{pmatrix} 0.8355 & 0.3932 & 0.4175 \\ 0.3927 & 0.8191 & 0.4255 \\ 0.3843 & 0.4177 & 0.8029 \end{pmatrix} \quad (4.45)$$

El índice de prestaciones en $m=11000$, equivalente a 1.375 segundos, tiene un valor de $PIA(t=11.000)=0.6901$, y la matriz de pesos W es:

$$W = \begin{pmatrix} 0.4021 & 0.4743 & 0.8287 \\ 0.7865 & 0.4466 & 0.3952 \\ 0.4687 & 0.7587 & 0.3964 \end{pmatrix} \quad (4.46)$$

En la Figura 4.15 se pueden ver las representaciones de las fuentes originales (Figura 4.15a) y las fuentes separadas (Figura 4.15b). Obsérvese que, en este caso, las señales reconstruidas van afectadas de un factor de escala, como se indicó en los Capítulos 1 y 2 (Sec.1.2 y 2.1.1). En la Figura 4.16 se muestra la variación en el tiempo del índice de prestaciones PIA, el error cuadrático y la diafonía.

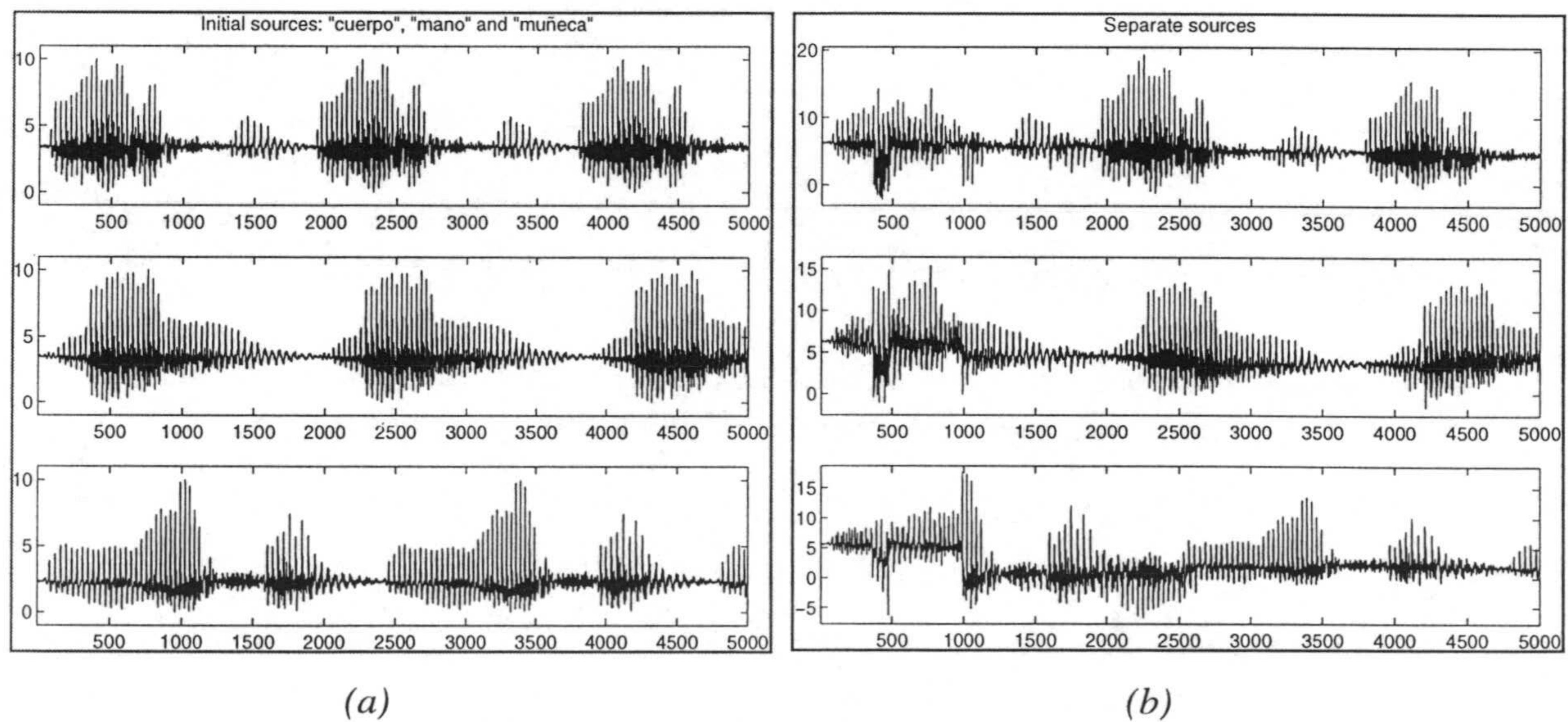
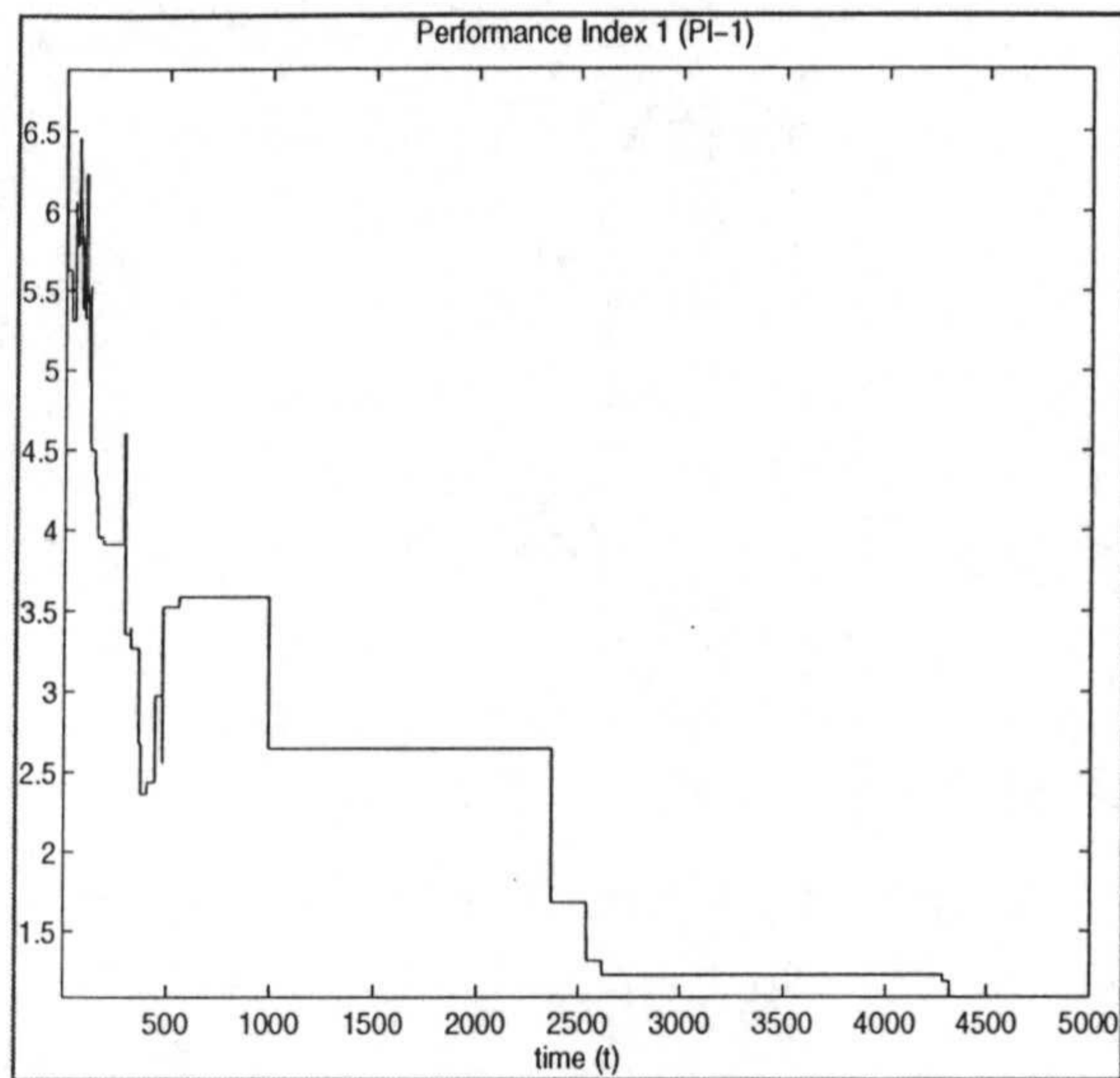
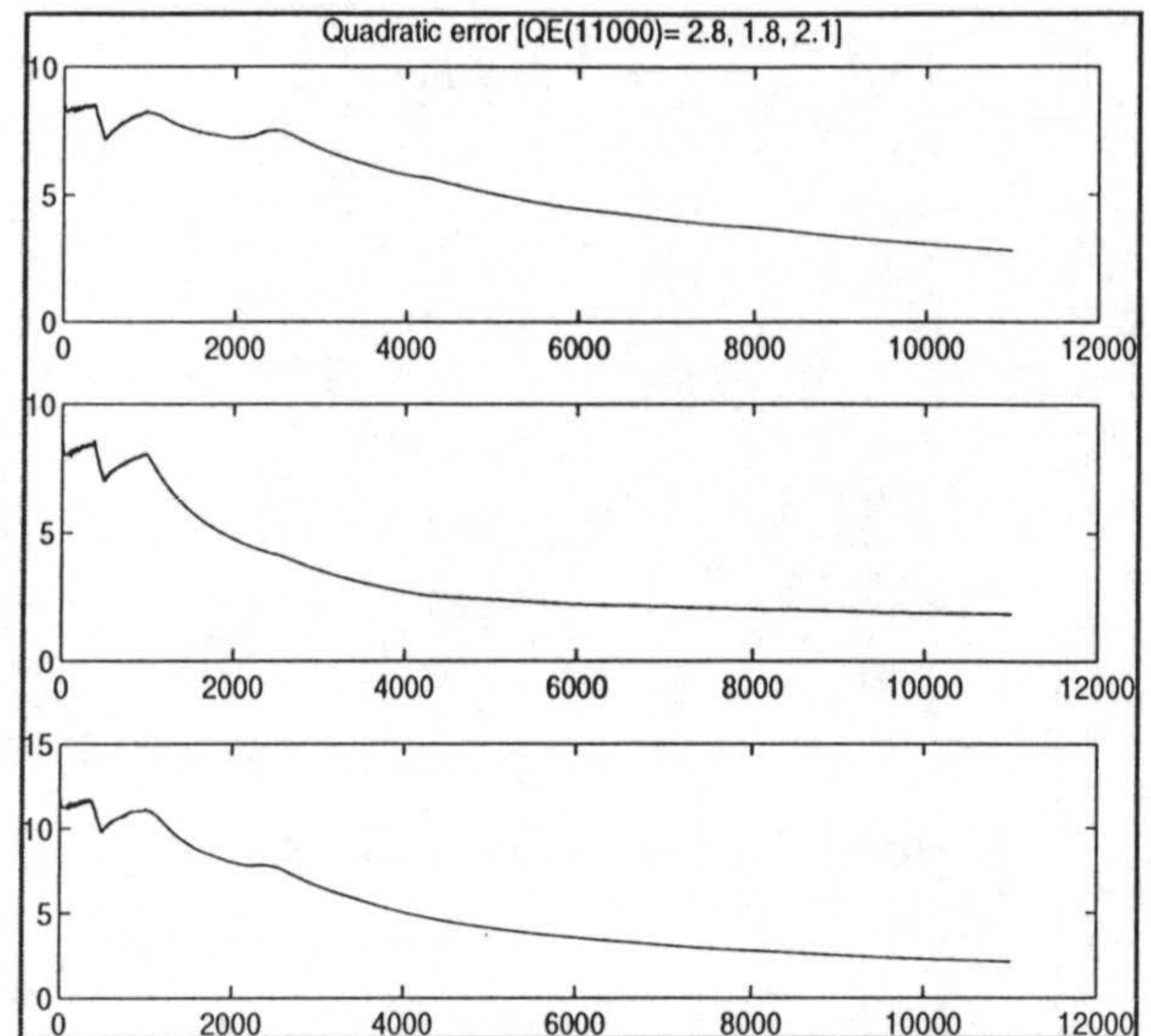


Figura 4.15. (a) señales originales, (b) señales separadas durante el proceso de aprendizaje.



(a)



(b)

Figura 4.16. Evolución durante el proceso de aprendizaje de (a) índice de prestaciones PIA, (b) la diafonía.

Para estudiar la convergencia del proceso iterativo de la reconstrucción de señales se ha utilizado este último ejemplo con distintos valores del parámetro de convergencia, ϵ . La Tabla 4.1 muestra el número medio de iteraciones necesario para obtener una diferencia entre valores sucesivos de y menor que ϵ . La **Figura 4.17** incluye el histograma de frecuencias para el caso $\epsilon=0.0001$.

Con los otros ejemplos se ha llevado a cabo un estudio similar. Los únicos problemas de convergencia surgen en algún caso al principio ($t < 1000$), cuando la red todavía no ha encontrado los pesos adecuados ($PIA > 4$).

Tabla 4.1 Número de iteraciones.

ϵ	Valor medio
0.1	6.35
0.01	9.07
0.001	11.36
0.00001	20.79

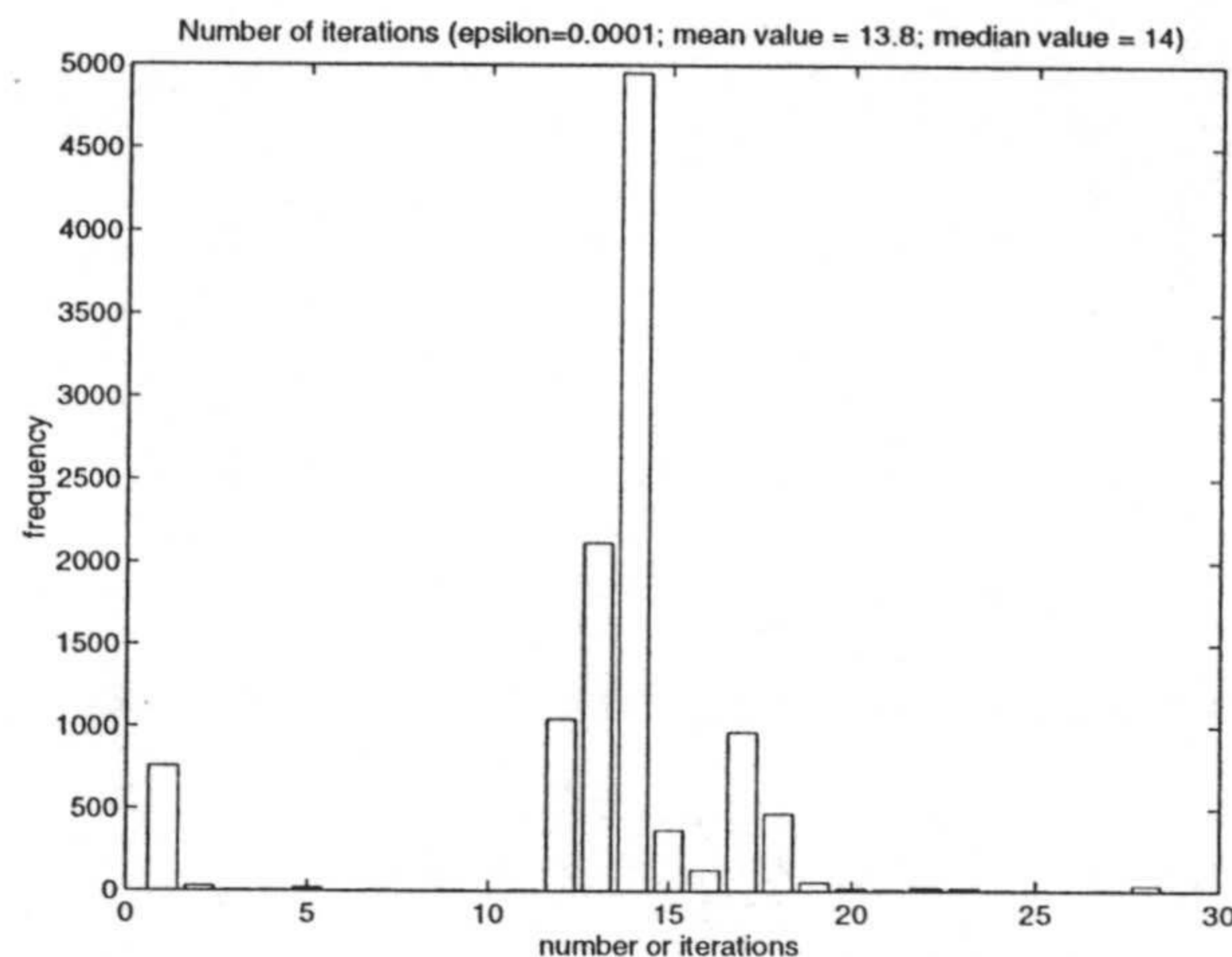


Figura 4.17. Histograma del número de iteraciones para reconstruir las señales.

Para estudiar el comportamiento de la red respecto al ruido se han llevado a cabo varias simulaciones usando las señales descritas en (4.38). En la Figura 4.18a y Figura 4.18b se puede observar la variación del índice de prestaciones de Amari (Sec.2.5.1) y del error cuadrático medio entre los elementos de la matriz \mathbf{W} , reordenada con respecto a los elementos de \mathbf{A} (Sec.2.5.2.1) durante el proceso de aprendizaje, sin ruido ($N=0$) y con distintos niveles de ruido introducido en la mezcla, obteniéndose señales corrompidas con relaciones señal/ruido de 20, 14, 10 y 8 dBs respectivamente. En todos los casos el ruido se ha modelado con una señal aleatoria uniformemente distribuida en el intervalo $[-N_0, N_0]$, donde N_0 viene determinado por la SNR correspondiente.

Los resultados obtenidos confirman que la traslación adaptativa del vértice compensa el ruido añadido, mientras que la evolución de las curvas en la Figura 4.18 muestran que con un incremento en el tiempo los índices de prestaciones tienden a aproximarse lentamente a los obtenidos sin añadir ruido. La Tabla 4.2 muestra las traslaciones automáticas realizadas para $t > 1000$ en los casos anteriormente citados.

Tabla 4.2. Traslaciones de las mezclas del ejemplo 2, valores de t en los que hay traslación.

Ruido nulo ($N=0$) $n_o=5; \tau=10000$	SNR=20dBs $n_o=5; \tau=10000$	SNR=14dBs $n_o=5; \tau=10000$	SNR=10dBs $n_o=5; \tau=10000$	SNR=8dBs $n_o=5; \tau=10000$
1009	1009	1012	1007	19318
6610	4210	3227	1011	
7009	7010	11810	4967	
		14611	6610	

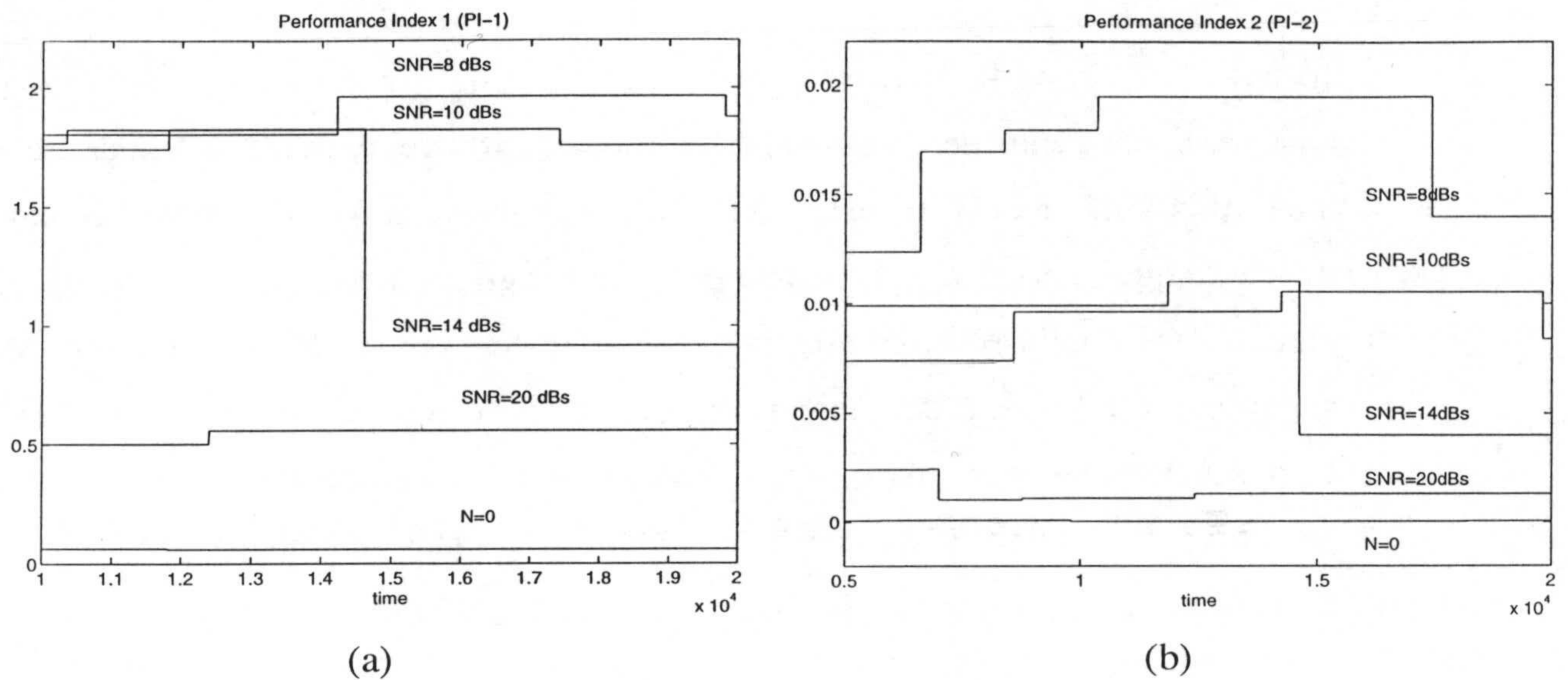


Figura 4.18. Evolución de los índices de prestaciones durante el proceso de aprendizaje con distintos niveles de ruido en las mezclas: (a) índice de Amari, (b) error cuadrático.

4.8 CONCLUSIONES

En este capítulo se ha presentado una red neuronal para obtener la separación ciega de mezclas lineales de p fuentes. Se han usado dos redes, una para el aprendizaje de los pesos (que se identifica con los coeficientes de la matriz de estimación de la mezcla) y otra para recuperar las fuentes originales. La red de aprendizaje actúa de forma no supervisada, y la de recuperación funciona de forma recursiva. Tanto el aprendizaje de los pesos como la separación se llevan a cabo simultáneamente, es decir, no hay dos fases distintas de aprendizaje y recuperación de las fuentes.

La limitación más importante de este procedimiento es que sólo se puede aplicar a medios lineales; además, para conseguir una buena separación es necesario obtener un vértice y p vectores en las aristas del espacio de observaciones. Esta última limitación tiene dos consecuencias: la convergencia depende de la probabilidad de que existan puntos próximos a las aristas del hiperparalelepípedo, y el algoritmo es sensible al ruido, produciendo un cierto error en la estimación de la matriz de mezcla. La primera consecuencia no es significativa, ya que en la práctica, con señales reales como voz, señales biomédicas o imágenes, se pueden conseguir, en tiempos razonables, las combinaciones que dan lugar a los vectores requeridos.

En el Ejemplo 3 de la Sec.4.7 se muestra que para señales de voz se obtiene una matriz de mezcla adecuada en menos de 2 segundos. En lo que se refiere al efecto del ruido, se ha demostrado (Sec.4.3) que si es uniforme y acotado a lo largo del medio, la única consecuencia es que se necesita más tiempo para obtener un resultado aceptable, ya que los coeficientes de los índices de prestaciones se obtienen como diferencia entre las observaciones, que a lo largo del tiempo compensan este efecto. Como se ha visto en la Sec.4.7 con señales sintéticas a las que se añadió un ruido blanco acotado con una SNR de 8 dBs, se obtuvo un error cuadrático medio entre los coeficientes de la matriz estimada y la matriz de mezcla menor de 0.015, tras haber procesado 17.000 muestras (ver Figura 4.18).

Otra característica del procedimiento es que se puede aplicar a señales estadísticamente dependientes, siempre que tal dependencia no implique la no generación de vectores que se correspondan con las p aristas del hiperparalelepípedo que converjen en el vértice de traslación (Sec.4.7.4, Ejemplo2).

También se han mostrado resultados experimentales con señales de voz con una convergencia más que aceptable de la red neuronal usada para reconstruir las fuentes originales.

CAPÍTULO 5:

ALGORITMO DE LOCALIZACIÓN DE LOS EJES DE LA DISTRIBUCIÓN DE VECTORES DE MEZCLA PARA SEPARACIÓN CIEGA DE SEÑALES.

En el capítulo anterior se presentó un algoritmo de separación cuya convergencia depende de las probabilidades de generación de valores extremos en las fuentes (vectores críticos). Esto provoca que el tiempo para obtener una matriz de recuperación crezca considerablemente cuando las funciones de densidad de probabilidad de las fuentes sean de tipo subgaussiano o subuniforme, o cuando del número de fuentes, p , crece.

En el presente capítulo se presenta otro método de separación, también basado en propiedades geométricas (Capítulo 2) que es especialmente válido para separar señales con funciones de densidad de probabilidad unimodales y centradas; es decir, con un máximo en el centro del rango de sus valores. Entre estas distribuciones se encuentran las funciones gaussiana, gamma bilateral, laplaciana (o doble exponencial), algunas beta (con $\alpha=\beta>1$), etc. (ver Apéndice 1). Diversos tipos de señales reales tienen funciones de densidad de probabilidad unimodales centradas, como pueden ser las señales de voz y música (señales

de audio) de gran importancia en procesamiento de señales, por lo que este capítulo se centra en este tipo de fuentes.

El capítulo se inicia (Sec.5.1) poniendo de manifiesto algunas de las propiedades de las señales de voz y de las señales unimodales centradas, así como la lenta convergencia que se obtienen con la aplicación a ellas del algoritmo de máxima separación angular.

La Sec.5.2 introduce las bases del procedimiento a partir del concepto de ejes de la distribución de los vectores de mezcla, y se indica que dichos ejes son paralelos al hiperparalelepípedo que contiene el espacio de observaciones. Por tanto, según se probó en el Capítulo 2, con las pendientes de los ejes de la distribución se puede estimar la matriz de mezcla.

En la Sec.5.3 se presentan dos algoritmos, uno adaptativo, pensado para funcionar en tiempo real, y otro, para actuar fuera de línea pero más sencillo de implementar.

En la Sec.5.4 se muestran los resultados experimentales obtenidos con distintas matrices de mezcla, y fuentes tanto sintéticas (Sec.5.4.1) como reales (Sec.5.4.2).

El procedimiento propuesto se basa en la clasificación angular de los vectores de mezcla, una vez realizado un cambio de coordenadas. La precisión de los resultados depende del número de clases considerado. Esta dependencia se estudia en la Sec.5.5.

El nuevo algoritmo no es *equivariante*, por lo que es conveniente discutir los resultados obtenidos en función de los valores de los elementos de la matriz de mezcla, llevándose esto a cabo en la Sec.5.6.

Por último en la Sec.5.7 se analizan los efectos del ruido en la convergencia del algoritmo.

5.1 SEPARACIÓN DE MEZCLAS DE SEÑALES DE VOZ

La densidad de probabilidad de las señales de voz se puede estimar sencillamente mediante un histograma de amplitudes, considerando un gran número de muestras. Se puede comprobar [RAB78] que una buena aproximación de la densidad de amplitudes de las señales de voz se obtiene mediante una función de distribución gamma bilateral (Sec. A1.3.3, Figura 5.1):

$$pr(s) = \left[\frac{\sqrt{3}}{8\pi\sigma_s |s|} \right]^{1/2} \cdot e^{-\frac{\sqrt{3}|s|}{2\sigma_s}} \quad (5.1)$$

donde $|s|$ representa el valor absoluto de la señal de voz, $s(t)$, y $(\sigma_s)^2$, su varianza. También, con frecuencia, la distribución de densidad de amplitudes de señales de voz se suele aproximar a una función de densidad de probabilidad laplaciana (Sec.A1.3.4, Figura A1.4) [RAB78].

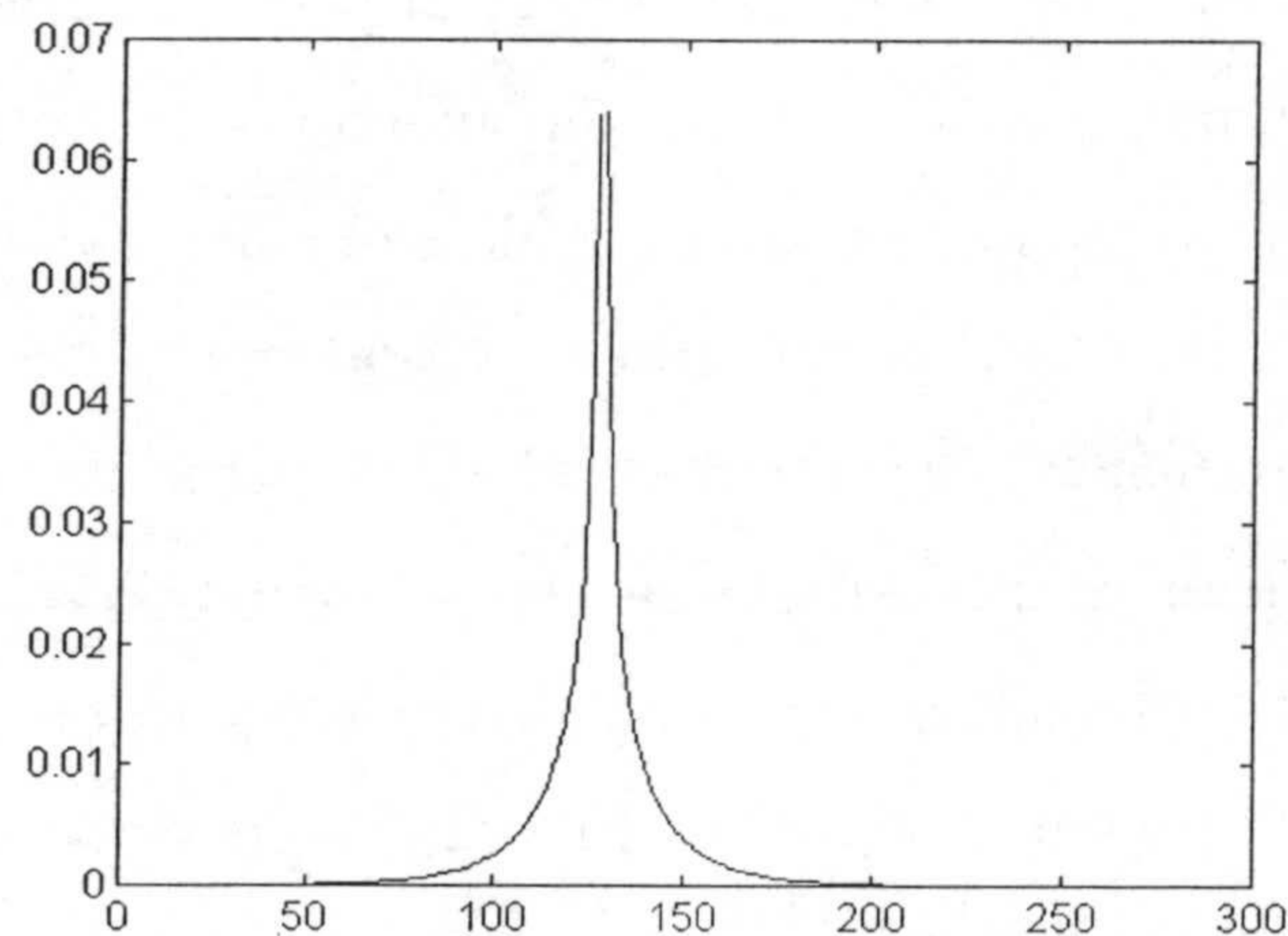


Figura 5. 1. *Función de distribución gamma.*

El procedimiento propuesto en este capítulo no sólo se puede aplicar a señales de voz. En efecto, las propiedades en que se basa el nuevo método, como se verá más adelante, en las Secciones 5.2 y 5.3, radican en que la función de distribución es unimodal (con un solo máximo) y centrada (con el máximo en el centro), de forma tal que todo lo indicado en las siguientes secciones se puede aplicar a señales cuyas funciones de distribución sean de este tipo, tales como doble exponencial (laplaciana), gamma bilateral y beta (con $\alpha=\beta > 1$) (ver Apéndice I).

Para poder separar las señales usando el algoritmo propuesto en el Capítulo 4, es necesario detectar al menos un vértice del hiperparalelepípedo del espacio de observaciones y un punto en cada una de las aristas que inciden en dicho vértice.

Teniendo en cuenta las densidades de amplitud de las señales de voz, la probabilidad de obtener un punto en una de las aristas es muy pequeña. Nótese (Capítulo 2), que los puntos en las *aristas* de los ejes en el espacio de observaciones son las imágenes de puntos en las *aristas* del espacio fuente, es decir, corresponden a la producción de valores extremos (máximos, s_{iM} y mínimos, s_{im}) de las señales s_i .

Se han ajustado señales de voz reales a la función de distribución gamma, como se muestra más adelante en la Sec.5.4.2 (Tabla 5.7) y se obtiene que la probabilidad de obtener valores extremos es del orden de 10^{-6} , lo que equivale a decir que, por término medio, cada 10^6 *muestras* $= 10^6 \cdot T_s = 10^6 \cdot 5.5 \cdot 10^{-5} = 45$ *segundos* (T_s es el periodo de muestreo de la señal de voz) podría obtenerse un punto en una arista, (suponiendo dos fuentes). La probabilidad de obtener un vértice es mucho menor, ya que equivale a obtener simultáneamente dos puntos en dos aristas adyacentes. Así, por ejemplo, suponiendo que las dos fuentes son estadísticamente independientes, la probabilidad de generar un vértice $s(s_M, s_m)$ sería $f_{s1}(s_m) \cdot f_{s2}(s_M) \approx 10^{-12}$; es decir, se necesitaría un tiempo medio de $45 \cdot 10^6 \text{s} \approx 1.43$ años para obtener dicho vértice. Estas conclusiones se pueden visualizar gráficamente representando el espacio de mezcla de las señales de voz. La Figura 5.2 muestra dos ejemplos de mezclas. Obsérvese que la distribución de los puntos de mezcla es tal que en las zonas más lejanas del centro apenas hay puntos. La probabilidad de obtener puntos en cualquiera de los vértices A, B, C o D es muy baja (Figura 5.2).

También hay que tener en cuenta que el procedimiento propuesto en el Capítulo 4 es muy sensible al ruido producido en el medio, ya que este puede desplazar los vértices y aristas del espacio de observaciones. La contribución del ruido que provoca el desplazamiento de un vértice se puede reducir considerando vectores de mezclas de las aristas que estén lejos del vértice en cuestión (Sec.4.3 y 4.6). Estos vectores corresponden a puntos del espacio de las fuentes que distan del centro de la distribución y que son muy poco probables (Figuras 5.2a y 5.2c). El tiempo de convergencia se ve considerablemente incrementado cuando se

trata de reducir la influencia del ruido en el método descrito en el Capítulo 4. El nuevo procedimiento propuesto en el presente capítulo reduce considerablemente este problema como se verá en la Sec.5.7.

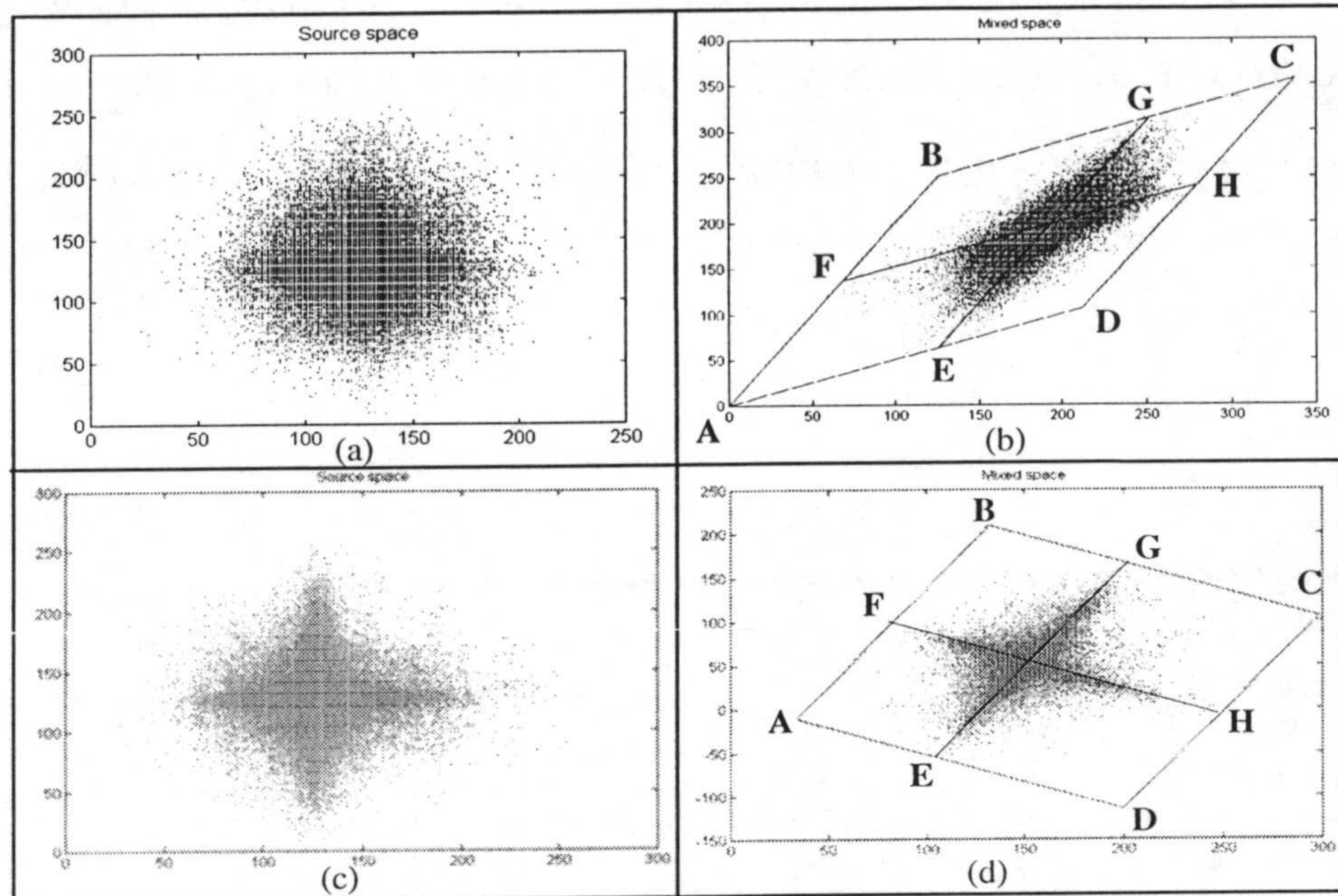


Figura 5. 2. Espacios fuente y de mezclas de señales de voz con:

$$(a), (b) \text{ Matriz } A_1 = \begin{pmatrix} 1 & 0.5 \\ 0.5 & 1 \end{pmatrix}; (c) \text{ y } (d) \text{ Matriz } A_3 = \begin{pmatrix} 0.8 & 0.4 \\ -0.5 & 0.9 \end{pmatrix}.$$

5.2 BASES DEL PROCEDIMIENTO

En la Figura 5.2 se puede apreciar que se cumplen las siguientes propiedades en el espacio de fuentes y de mezclas para señales con función de distribución unimodal y centrada:

Propiedad 1. La máxima probabilidad de ocurrencia de vectores de mezcla corresponde al centro del hiperparalelepípedo. Esta es una consecuencia obvia de la linealidad de la transformación A . Además, como se deduce de los conceptos incluidos en la Sec.2.2, el centro del hiperparalelepípedo del espacio fuente tiene como imagen el centro del espacio de mezcla. Como las señales de voz originales están centradas, con el máximo de probabilidad en este centro, el mayor número de vectores e se generarán en el centro, O^* .

Propiedad 2. Debido a la linealidad de la transformación A , la mezcla se distribuye con un máximo de probabilidad en las direcciones paralelas a las aristas del

hiperparalelepípedo del espacio de observaciones (direcciones EG y FH en la Figura 5.2). Denominamos a dichas direcciones *ejes de la distribución*.

Para comprobar lo dicho en el párrafo anterior, la Figura 5.3a representa la distribución (histograma) de los ángulos formados por los vectores $e(t)$ trasladados al centro O^* ($e^*(t)$) con respecto a la horizontal (eje e_1^*). Esta figura muestra que los máximos de la distribución se obtienen en los ángulos $\beta_1=26.6^\circ$ y $\beta_2=63.4^\circ$, que corresponden a las pendientes:

$$\arctg(\beta_1) = 0.5 \quad \arctg(\beta_2) = 2 \quad (5.2)$$

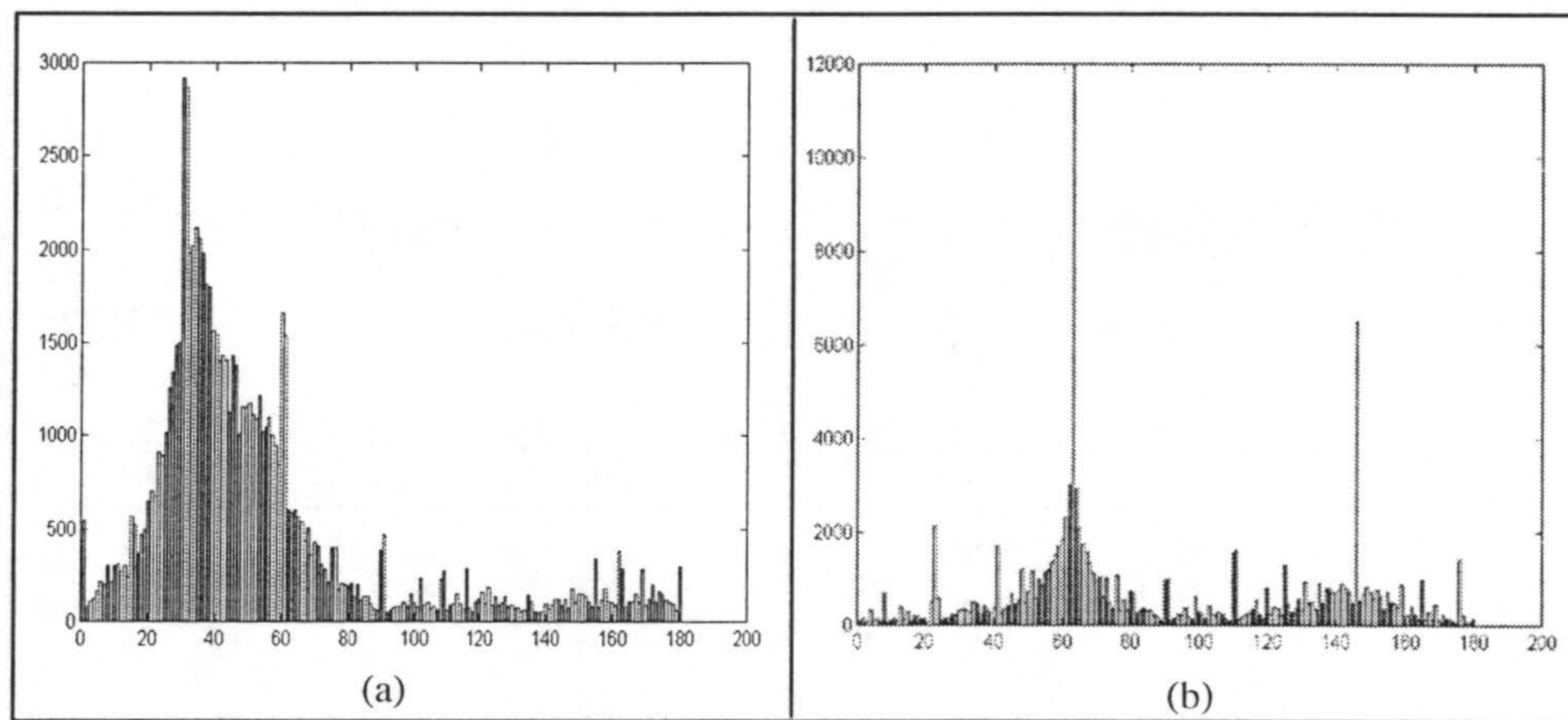


Figura 5. 3. Distribución de frecuencias angulares de mezclas con:

$$(a) \text{ Matriz } A_1 = \begin{pmatrix} 1 & 0.5 \\ 0.5 & 1 \end{pmatrix}; (b) \text{ Matriz } A_3 = \begin{pmatrix} 0.8 & 0.4 \\ -0.5 & 0.9 \end{pmatrix}.$$

Estos valores coinciden con las pendientes (sl) de las aristas del hiperparalelepípedo del espacio de mezclas ($A_1 = \begin{pmatrix} 1 & 0.5 \\ 0.5 & 1 \end{pmatrix}$), que vienen dadas por (Sec.2.2):

$$sl_{01} = \frac{a_{21}}{a_{11}} = \frac{0.5}{1} = 0.5 \quad ; \quad sl_{02} = \frac{a_{22}}{a_{12}} = \frac{1}{0.5} = 2 \quad (5.3)$$

Se llega a la misma conclusión a partir de otras matrices de mezcla y señales de voz; así los máximos en la Figura 5.3b se obtienen en 66° y 148° , que coincide con las pendientes

$$\text{de la matriz } A_3 = \begin{pmatrix} 0.8 & 0.4 \\ -0.5 & 0.9 \end{pmatrix}.$$

De acuerdo con lo indicado en párrafos anteriores, el problema de separación de fuentes, para el tipo de señales descrito, se reduce a encontrar los ejes de la distribución de probabilidades en el espacio de mezclas, ya que estos ejes son paralelos a los ejes del hiperparalelepípedo del espacio de observaciones. Es interesante notar que:

1. Las conclusiones anteriores se pueden aplicar no sólo a señales con distribución gamma, sino también a todos los tipos de fuentes que tengan una distribución con una única moda o máximo (unimodales) y centrada.
2. El método presenta una inmunidad razonable al ruido blanco. En efecto, al ser el ruido blanco, éste desplaza los puntos de la mezcla aleatoriamente *alrededor* de los puntos producidos si no hubiese ruido, y con media cero, dejando el centro (que se obtiene como media) inalterado y los ejes de la distribución sólo ligeramente modificados (quedan *difuminados*) (Sec.5.7). Recuérdese que el procedimiento propuesto en este capítulo estima los elementos de la matriz \mathbf{A} , a partir de los ejes de la distribución.

5.3 ALGORITMOS PARA SEPARACIÓN DE SEÑALES UNIMODALES CENTRADAS.

Como se describió en la sección anterior, el problema se reduce a obtener los ejes de distribución en el espacio de mezclas. En la práctica, se pueden considerar varias alternativas para llegar a este objetivo.

En primer lugar (Sec. 5.3.1) se describirá un algoritmo adaptativo (*en línea*), proyectado para funcionar en tiempo real. Posteriormente (Sec. 5.3.2) se realizarán algunas consideraciones sobre la implementación del algoritmo, y por último, (Sec. 5.3.3) se propone un algoritmo que funciona de forma *batch* (*fuera de línea*).

5.3.1 Algoritmo adaptativo para separar mezclas lineales de señales con distribuciones unimodales centradas.

En esta sección se propone un algoritmo sencillo que se puede implementar eficientemente en un DSP u otro hardware específico para llevar a cabo la separación de fuentes en tiempo real. Este procedimiento, aplicado a señales de audio, supone que las operaciones que se realizan en cada iteración se efectúan en un tiempo inferior a $T=45.35\mu s$, que es el periodo necesario para obtener señales de voz con calidad "radio".

Por sencillez, se ha implementado el algoritmo para $p=2$, pero se puede generalizar a más fuentes. La finalidad es obtener las direcciones de los ejes de distribución.

El procedimiento puede comprenderse fácilmente con ayuda de la Figura 5.4. Para obtener las direcciones de los ejes de la distribución del espacio de mezclas se toma como referencia el centro del hiperparalelepípedo $\mathbf{O}^* = (e_{10}, e_{20})$. El espacio \mathbf{e}^* , tomando como centro \mathbf{O}^* , se divide en $2 \cdot n_c$ sectores o clases (*clusters*) diferentes, de forma que cada sector corresponde a un ángulo de $2\pi/(2n_c)$ Radianes. Los ejes de la distribución se obtienen determinando dos clases (c_{M1} y c_{M2}) en las que se produce un máximo relativo en la frecuencia de los vectores de mezcla. Esto se lleva a cabo de la siguiente forma:

Para cada instante $t=mT$, $m=1, \dots, M$:

- Se toma un vector $\mathbf{e}(t)=(e_1(t), e_2(t))$
- Se hace un cambio de coordenadas: $\mathbf{e}^*(t)=\mathbf{e}(t)-\mathbf{O}^*$,
- Se normaliza $\mathbf{e}^*(t)$:

$$\hat{\mathbf{e}}^* = \mathbf{e}^* / |\mathbf{e}^*| \quad \forall \mathbf{e}^* \mid |\mathbf{e}^*| \neq 0 \quad (5.4)$$

- Se actualiza la frecuencia de incidencias en la clase a la que pertenece el vector, $\hat{\mathbf{e}}^*$.
- Las dos clases (c_{M1} y c_{M2}) con frecuencia relativa mayor corresponden a las clases donde se encuentran los ejes de la distribución.

Cada clase se representa con un vector, \mathbf{e}_c^* , que corresponde a los ejes de la distribución.

Se propone que el vector representante de cada clase sea el vector media de todos los

vectores, \hat{e}_c^* , que inciden en dicha clase. Los vectores que representan a las dos clases con mayor distribución de frecuencias (los dos máximos relativos de la distribución de frecuencias angulares $(\hat{e}_{cM1}^*, \hat{e}_{cM2}^*)$) estarán en las direcciones de los ejes de distribución, y por tanto, sus componentes proporcionan las columnas de una matriz W similar a A .

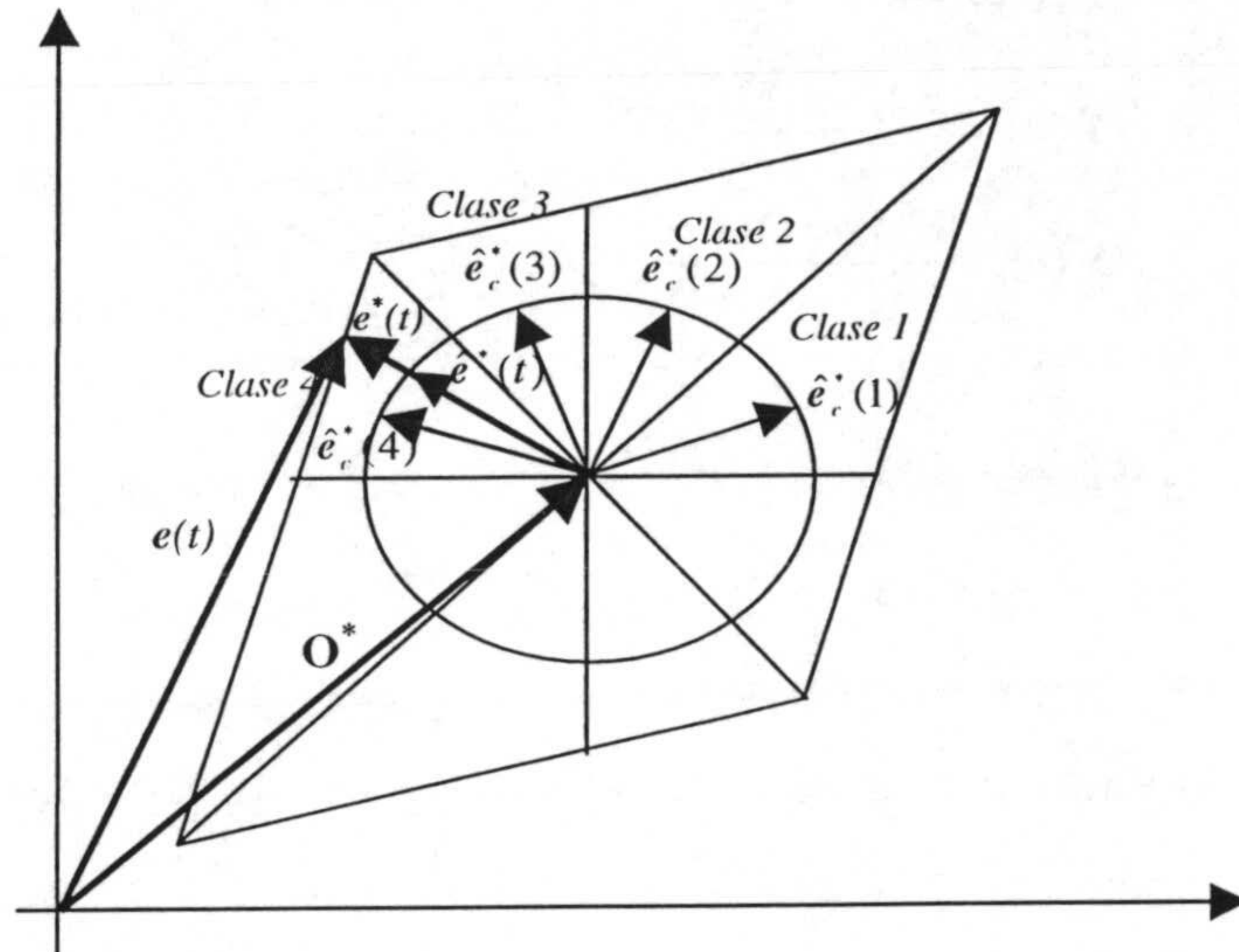


Figura 5. 4. Obtención de los ejes de distribución.

Para obtener adaptativamente el centro, O^* , se usa la siguiente expresión:

$$O^*(n) = O^*(n-1) + \alpha(n) \cdot [e(n) - O^*(n-1)] \quad (5.5)$$

El vector representante de cada clase, e_c^* , se determina de forma similar:

$$e_c^*(n) = e_c^*(n-1) + \alpha(n) \cdot [e(n) - e_c^*(n-1)] \quad (5.6)$$

$\forall e(n) \in \text{cluster } c$

$$y \quad \hat{e}_c^* = \frac{e_c^*}{|e_c^*|}$$

donde en las dos expresiones anteriores:

$$t = m \cdot T_s; \quad m = 1, \dots, M; \quad \alpha(m) = \frac{1}{m} \quad (5.7)$$

siendo T_s el periodo de muestreo.

Es interesante notar que las expresiones anteriores se han obtenido como fórmula recursiva para calcular el vector medio. Por ejemplo, para obtener el centro se tiene que:

$$\begin{aligned}
 \mathbf{O}^*(1) &= \frac{\mathbf{e}(1)}{1} \\
 \mathbf{O}^*(2) &= \frac{\mathbf{e}(1) + \mathbf{e}(2)}{2} = \mathbf{O}^*(1) \cdot \frac{1}{2} + \frac{\mathbf{e}(2)}{2} \\
 \mathbf{O}^*(3) &= \frac{\mathbf{e}(1) + \mathbf{e}(2) + \mathbf{e}(3)}{3} = \mathbf{O}^*(2) \cdot \frac{2}{3} + \frac{\mathbf{e}(3)}{3} \\
 &\dots \\
 \mathbf{O}^*(n) &= \mathbf{O}^*(n-1) \cdot \frac{n-1}{n} + \frac{\mathbf{e}(n)}{n} = \\
 &= \mathbf{O}^*(n-1) + \alpha(n) \cdot [\mathbf{e}(n) - \mathbf{O}^*(n-1)]
 \end{aligned} \tag{5.8}$$

Las expresiones (5.5) y (5.6) se pueden interpretar, geoméricamente, dentro del contexto del aprendizaje competitivo [RUM86], [KOH90]. En efecto, geoméricamente (5.5) indica que el vector representante del centro de cada clase (\mathbf{e}_c^*) gira hacia el nuevo vector ($\hat{\mathbf{e}}^*(n)$) una cantidad proporcional a la diferencia entre los dos vectores (Figura 5.5). El factor de proporcionalidad de este desplazamiento es $\alpha(m)$, que representa la velocidad de aprendizaje. Como se indica en el algoritmo de aprendizaje competitivo [KOH90], $\alpha(m)$ debe ser un coeficiente escalar monótonamente decreciente adecuado, $0 < \alpha(m) < 1$. En el presente caso la función escogida (5.7), $\alpha = 1/m$, satisface esta condición.

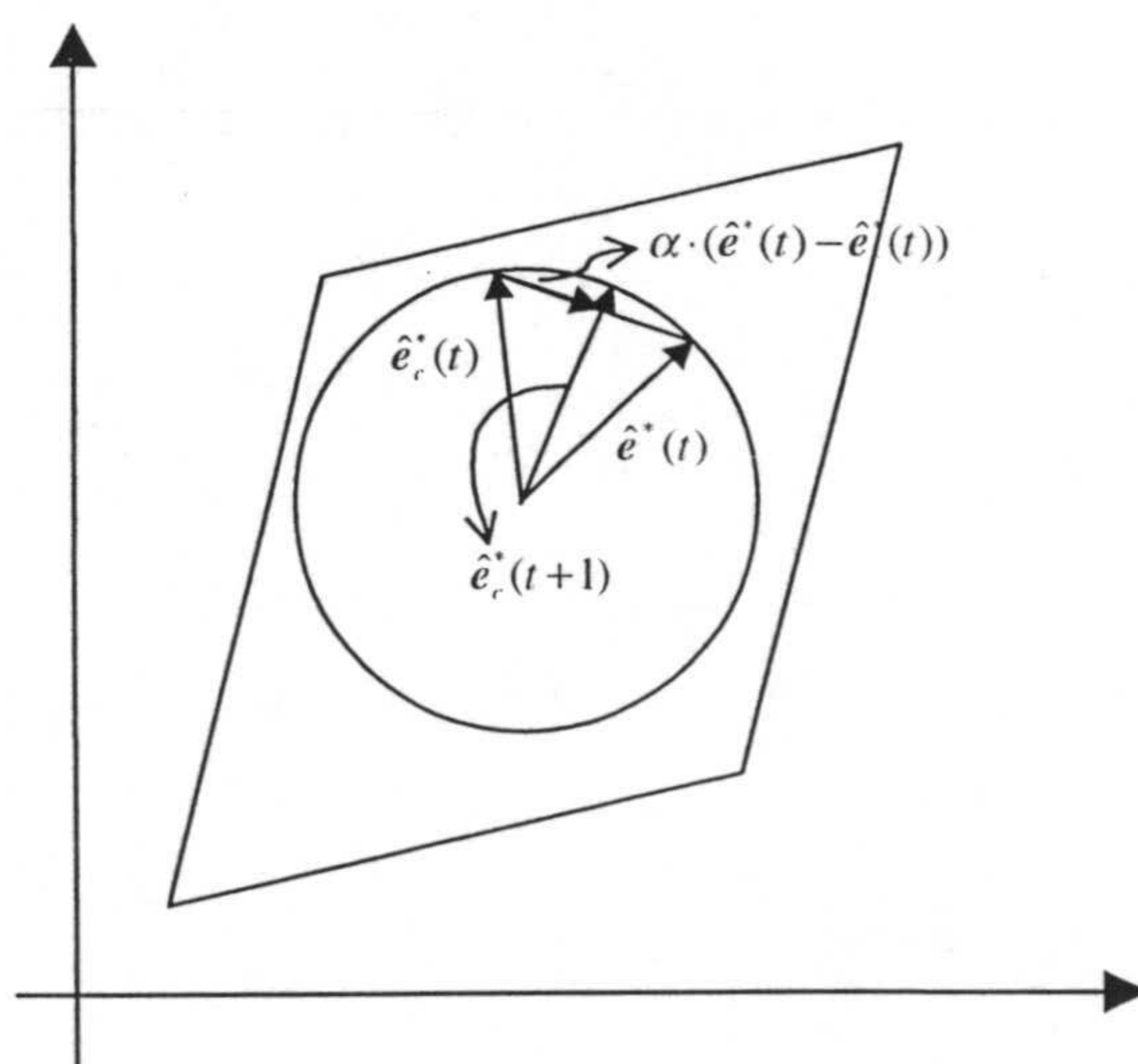


Figura 5.5. Interpretación de las expresiones (5.8) y (5.9).

La Tabla 5.1 resume los principales pasos del algoritmo implementado.

Tabla 5. 1. Algoritmo para estimar A.

<ul style="list-style-type: none"> • Inicializar los vectores representativos de cada clase, e_c^*, $c \in \{1, \dots, n_c\}$ • Inicializar el centro: $O^* \leftarrow e(1)$
<p>$n=2:NP$</p> <ol style="list-style-type: none"> 1. Actualizar el centro, O^*, (Ecuación 5.5) 2. Obtener los vectores de mezcla normalizados referidos al centro: $\hat{e}^*(t)$, (Ecuación 5.4) 3. Determinar la clase de $\hat{e}^*(t)$, c. Incrementar la frecuencia de los vectores que inciden en esa clase, $f_c \leftarrow f_c + 1$. Actualizar el vector representativo de la clase, e_c^*, (Ecuación 5.6). Normalizar e_c^*, (Ecuación 5.4). 5. Obtener las clases que contienen los dos máximos relativos, c_{M1} y c_{M2}. Los vectores buscados son: $w_{M1} = \hat{e}_{c_{M1}}^*$, $w_{M2} = \hat{e}_{c_{M2}}^*$; $w = [w_{M1} \ w_{M2}]$ <p>fin</p>

5.3.2 Implementación del algoritmo

Se ha implementado el algoritmo dentro del entorno de desarrollo descrito en el Capítulo 3, simulando que los cálculos se tienen que realizar en tiempo real, tal como se ha definido en el algoritmo descrito en la sección anterior.

El programa que implementa el algoritmo (“disa_sf”) se ha realizado siguiendo los pasos de la Tabla 5.1 apareciendo al final representados en una misma figura (Figura 5.6) los vectores e así como las dos clases ganadoras visualizándose las dos direcciones correspondientes a la mayor frecuencia de puntos.

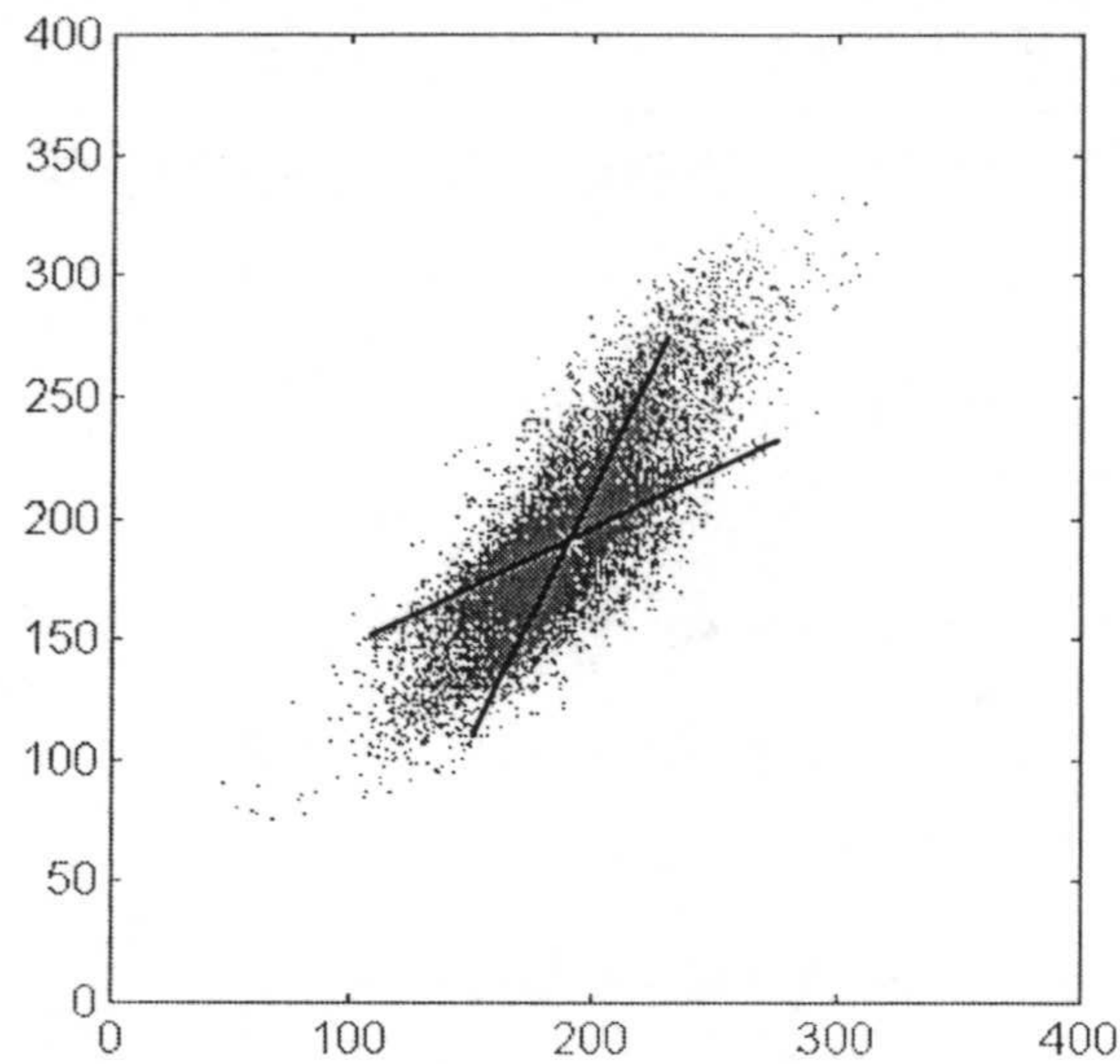


Figura 5.6. Ejes de la distribución de los vectores e .

También aparece simultáneamente una ventana con las matrices A y W en sus formas canónicas para poder comparar los resultados obtenidos. Al pulsar cualquier tecla aparece un nuevo gráfico con un histograma de frecuencias angulares (Figura 5.7a) en el que se pueden observar los dos máximos relativos correspondientes a las direcciones de la distribución.

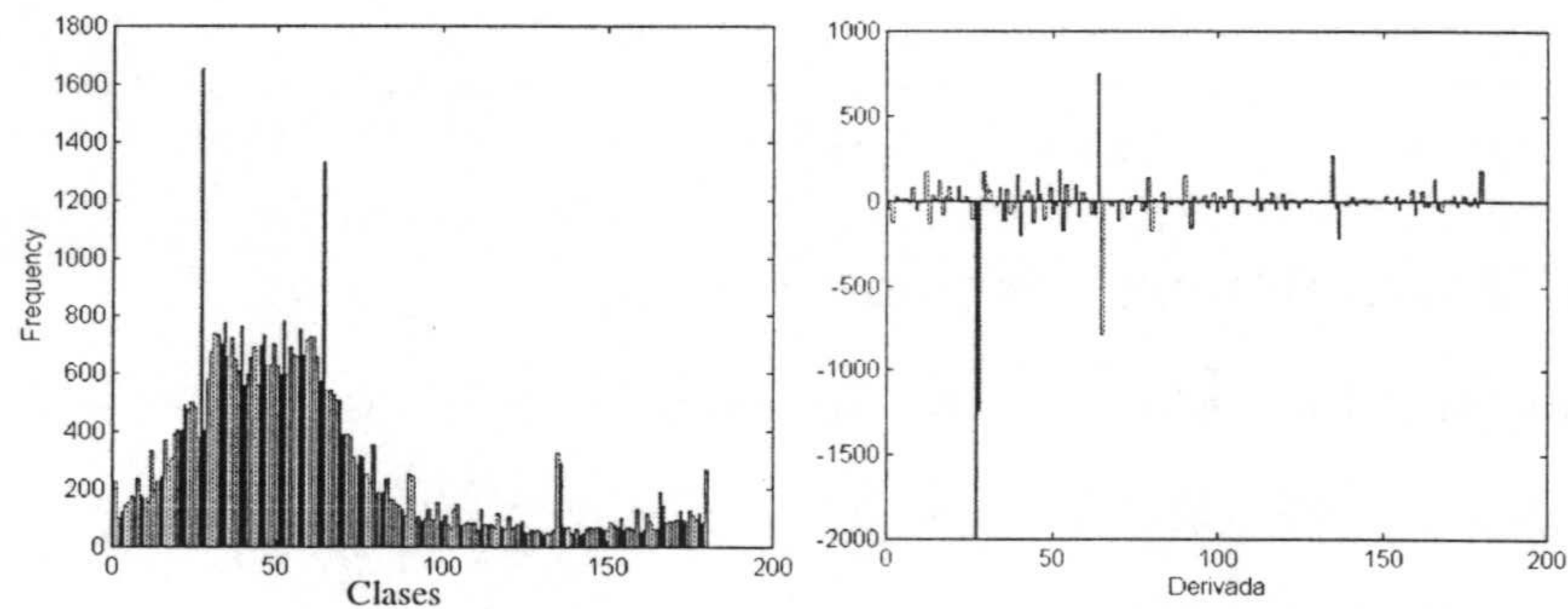


Figura 5. 7. Histograma de frecuencias y derivada.

Se han realizado dos versiones distintas del programa, diferenciándose una de otra en la forma de obtener los dos máximos relativos de la distribución de frecuencias angulares.

La primera versión (programa "disa_sf6.m") sencillamente utiliza el concepto de derivada. Se obtiene numéricamente la derivada sin más que restar al valor de cada clase

el valor de la clase anterior. Empíricamente en todos los casos se ha obtenido que los dos máximos relativos se producen en las clases cuya derivada primera es mayor (ver Figuras 5.7b, 5.14d, 5.15d y 5.19d). Para obtener mayor precisión se realiza lo siguiente:

1. Se obtiene la clase donde se produce el máximo absoluto, c_M .
2. Se obtiene la derivada primera de la distribución. En esta nueva distribución quedan mucho más resaltadas las posiciones de los dos máximos buscados.
3. Se elimina, en la derivada, la clase c_M con el máximo absoluto (el programa hace este valor igual a -2000, ver Figura 5.7b, 5.14d y 5.15d).
4. Se obtiene la posición del nuevo máximo absoluto de la derivada, c_M^* (67 en la Figura 5.5b).
5. Se determina la clase que denominamos *clase frontera*, de forma que:

$$c_f = \frac{c_M + c_M^*}{2} \quad (5.9)$$

6. Se obtienen los máximos absolutos en la distribución original en los intervalos de las clases $[1, c_f]$ y $[c_f+1, n_c]$, respectivamente. Las clases correspondientes a estos máximos, c_{M1} y c_{M2} , son las clases buscadas.

La segunda versión (programa "disa_sf8.m", ver Apéndice 3) utiliza otros conceptos para obtener la clase frontera de las dos regiones donde buscar los máximos absolutos y resulta más eficiente. A continuación se describe.

En la Figura 5.8 se observan las distintas formas de distribuciones angulares obtenidas empíricamente. El objetivo es encontrar la clase frontera, c_f , tal que los máximos absolutos de las regiones $[1, c_f]$ y $[c_f+1, n_c]$ sean los máximos relativos buscados (c_{M1} y c_{M2}). Se parte de la obtención del máximo, c_M , y el mínimo, c_m , absolutos de la distribución, y de la hipótesis de que los dos mínimos relativos de la distribución son equidistantes de los máximos relativos, tal y como se puede observar en la Figura 5.8. Esta hipótesis no es completamente real ya que las dos colinas son simétricas sólo aproximadamente, pero esta aproximación no presenta problemas ya que únicamente se pretende obtener la clase frontera como paso intermedio a la determinación de los dos intervalos donde obtener los máximos absolutos. Lo único que se requiere obtener con precisión son las clases donde se producen los máximos relativos, cosa que se efectúa en

una fase posterior. Debido a que la distribución se refiere a ángulos, de forma que $\text{ángulo} = \text{ángulo} + 2\pi$ en la Figura 5.8 se verificará que:

$$\begin{aligned} a &= a_1 + a_2 \\ b &= b_1 + b_2 \end{aligned} \quad (5.10)$$

También se verificará que:

$$n_c = 2a + 2b \quad (5.11)$$

o:

$$\begin{aligned} a &= \frac{n_c}{2} - b \\ b &= \frac{n_c}{2} - a \end{aligned} \quad (5.12)$$

La clase frontera, para los 6 casos considerados, se puede obtener en función de n_c , n_x y n_m (todos ellos conocidos):

$$\text{Caso a: } c_f = (c_m + a + b)_{\text{mod } n_c} = \left(c_m + \frac{n_c}{2}\right)_{\text{mod } n_c}$$

$$\text{Caso b: } c_f = (c_m + a + b)_{\text{mod } n_c} = \left(c_m + \frac{n_c}{2}\right)_{\text{mod } n_c}$$

$$\text{Caso c: } c_f = (c_m + a + b)_{\text{mod } n_c} = \left(c_m + \frac{n_c}{2}\right)_{\text{mod } n_c}$$

$$\text{Caso d: } c_f = c_m$$

$$\text{Caso e: } c_f = c_m$$

$$\text{Caso f: } c_f = (c_m + a + b)_{\text{mod } n_c} = \left(c_m + \frac{n_c}{2}\right)_{\text{mod } n_c}$$

Los casos (d) y (e) pueden identificarse teniendo en cuenta que en ellos, se verifica simultáneamente que:

$$c_m > a \text{ y } c_m < a + 2b = n_c - a \quad (5.13)$$

siendo $a = |c_M - c_m|$.

En definitiva, el algoritmo para la obtención de n_f es:

Hacer $c_f = c_m + \frac{n_c}{2}$
 Si $a < c_m < n_c - a$ hacer $c_f = c_m$

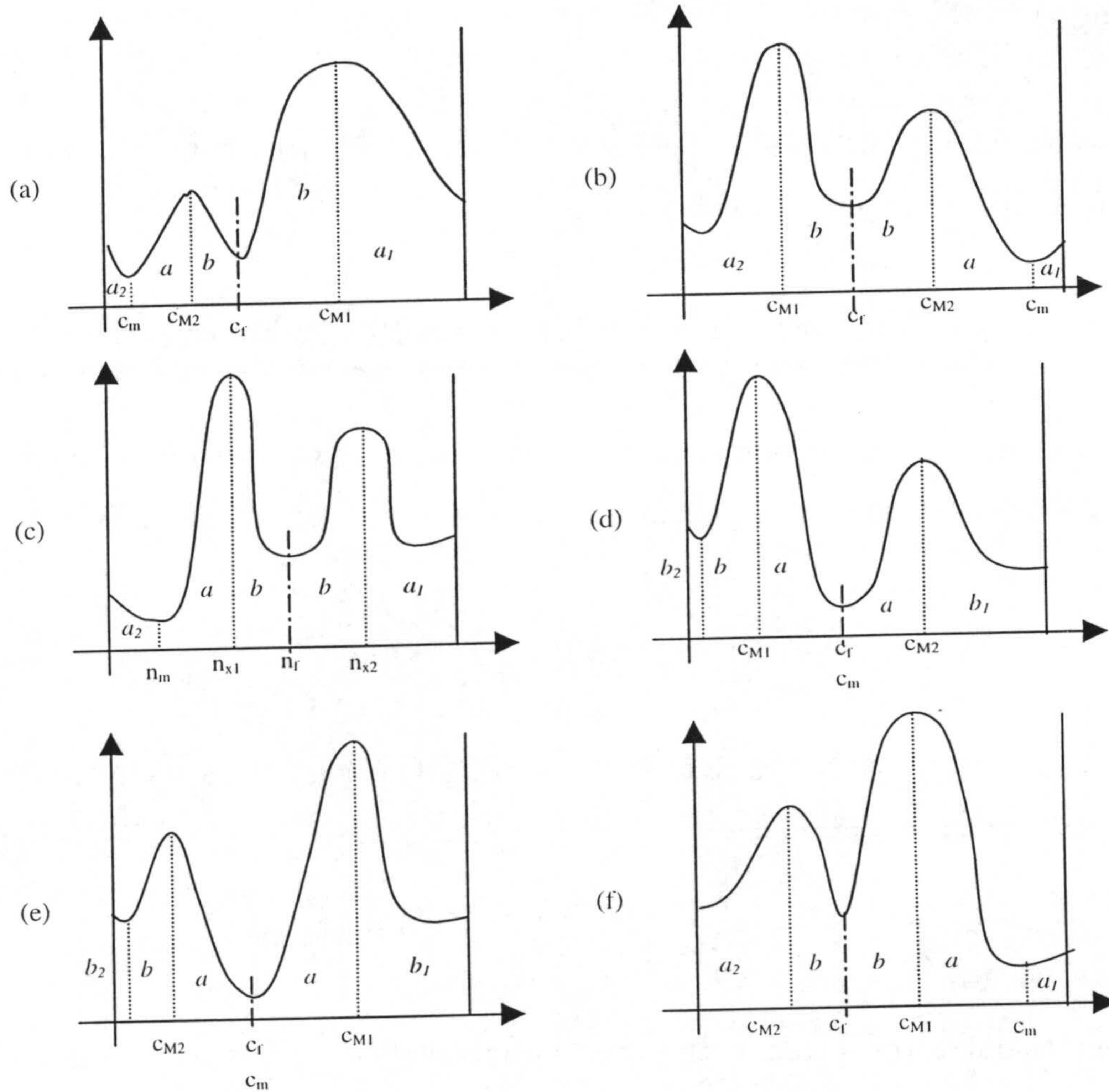


Figura 5. 8. Formas posibles de distribuciones de frecuencias angulares.

5.3.3 Algoritmo fuera de línea para separar mezclas lineales de señales con distribuciones unimodales centradas.

A diferencia del algoritmo descrito en la Sec. 5.3.1 que realiza el proceso muestra a muestra estimando adaptativamente la matriz de mezcla, el algoritmo que se propone en la presente sección opera con bloques de datos, ya que considera globalmente un total de

M muestras, no obteniéndose, por tanto, resultados hasta transcurrido un tiempo dado por:

$$T_{\text{resultado}} = M \cdot T_s + T_c \quad (5.14)$$

donde T_s es el periodo de muestreo y T_c es el tiempo de cómputo consumido por el algoritmo.

Al disponer en el momento de aplicar el algoritmo de la totalidad de las muestras, el procedimiento se simplifica notablemente.

Las bases del procedimiento son las mismas que las del algoritmo adaptativo.

Ahora el centro del hiperparalelepípedo se puede obtener sin más que estimar los valores esperados (medias) de las componentes de los vectores de observación. Es decir:

$$O^* = (E\{e_1\}, E\{e_2\}) = (\langle e_1 \rangle, \langle e_2 \rangle) \quad (5.15)$$

Referir cada vector e al centro se efectúa sin más que restar a cada vector de observación, $e(t)$, el vector centro, O^* :

$$e^*(t) = e(t) - O^* \quad (5.16)$$

Los ángulos que forman cada vector con la horizontal (eje e_1^+) se obtienen a partir de las componentes de $e^*(t)$. Utilizando un entorno de programación que disponga de la función arcotangente, los ángulos se calcularán como:

$$\text{angulo}(t) = \text{arctg} \frac{e_2^*(t)}{e_1^*(t)} \quad (5.17)$$

Posteriormente se obtienen las frecuencias de distribución de estos ángulos considerando un total de n_c clases.

Este algoritmo se ha simulado en MATLAB (programa "prue_dis.m"). Las frecuencias de cada clase se han calculado con la función *hist* de MATLAB y los dos máximos relativos se obtienen con el procedimiento descrito en la Sec. 5.3.2.

5.4 RESULTADOS EXPERIMENTALES.

Para considerar distintos tipos de matrices de mezcla que produzcan diversidad de ángulos de separación entre los ejes de la distribución se han seleccionado las 8 matrices de la Tabla 5.2. Los resultados experimentales que se presentan en el resto del capítulo corresponden a dichas matrices de mezcla.

Tabla 5.2. Matrices de mezcla usadas en este capítulo.

Matriz A	$A_1 = \begin{pmatrix} 1 & 0.5 \\ 0.5 & 1 \end{pmatrix}$	$A_2 = \begin{pmatrix} 0.8 & 0.1 \\ 0.2 & 0.9 \end{pmatrix}$	$A_3 = \begin{pmatrix} 0.8 & 0.4 \\ -0.5 & 0.9 \end{pmatrix}$	$A_4 = \begin{pmatrix} 0.3 & -0.1 \\ -0.4 & -1 \end{pmatrix}$
Matriz A canónica	$A_{d1} = \begin{pmatrix} 1 & 0.5 \\ 0.5 & 1 \end{pmatrix}$	$A_{d2} = \begin{pmatrix} 1 & 0.1 \\ 0.25 & 1 \end{pmatrix}$	$A_{d3} = \begin{pmatrix} 1 & 0.4 \\ -0.62 & 1 \end{pmatrix}$	$A_{d4} = \begin{pmatrix} 1 & -0.1 \\ -1.3 & -1 \end{pmatrix}$
Matriz A	$A_5 = \begin{pmatrix} -0.7 & 0.1 \\ 0.1 & -1 \end{pmatrix}$	$A_6 = \begin{pmatrix} -1 & -0.1 \\ -0.8 & 1 \end{pmatrix}$	$A_7 = \begin{pmatrix} 0.6 & 0.7 \\ -0.3 & 1.4 \end{pmatrix}$	$A_8 = \begin{pmatrix} 0.8 & -0.88 \\ 0.77 & 0.9 \end{pmatrix}$
Matriz A canónica	$A_{d5} = \begin{pmatrix} 1 & -0.1 \\ -0.14 & 1 \end{pmatrix}$	$A_{d6} = \begin{pmatrix} 1 & -0.1 \\ 0.8 & 1 \end{pmatrix}$	$A_{d7} = \begin{pmatrix} 1 & 0.5 \\ -0.5 & 1 \end{pmatrix}$	$A_{d8} = \begin{pmatrix} 1 & -0.98 \\ 0.97 & 1 \end{pmatrix}$

Como se dijo en la Sección 5.3, a partir de la distribución (histograma) de los ángulos formados por los vectores $e(t)$ trasladados al centro O^* ($e^*(t)$) con respecto a la horizontal (eje e_1^+), se pueden calcular los máximos de dicha distribución, que corresponden a las pendientes (sl) de las aristas del hiperparalelepípedo del espacio de mezclas, que según se justifica en la Sección 2.2, vienen dadas por:

$$sl_{01} = \frac{a_{21}}{a_{11}} \quad (5.18)$$

$$sl_{02} = \frac{a_{22}}{a_{12}}$$

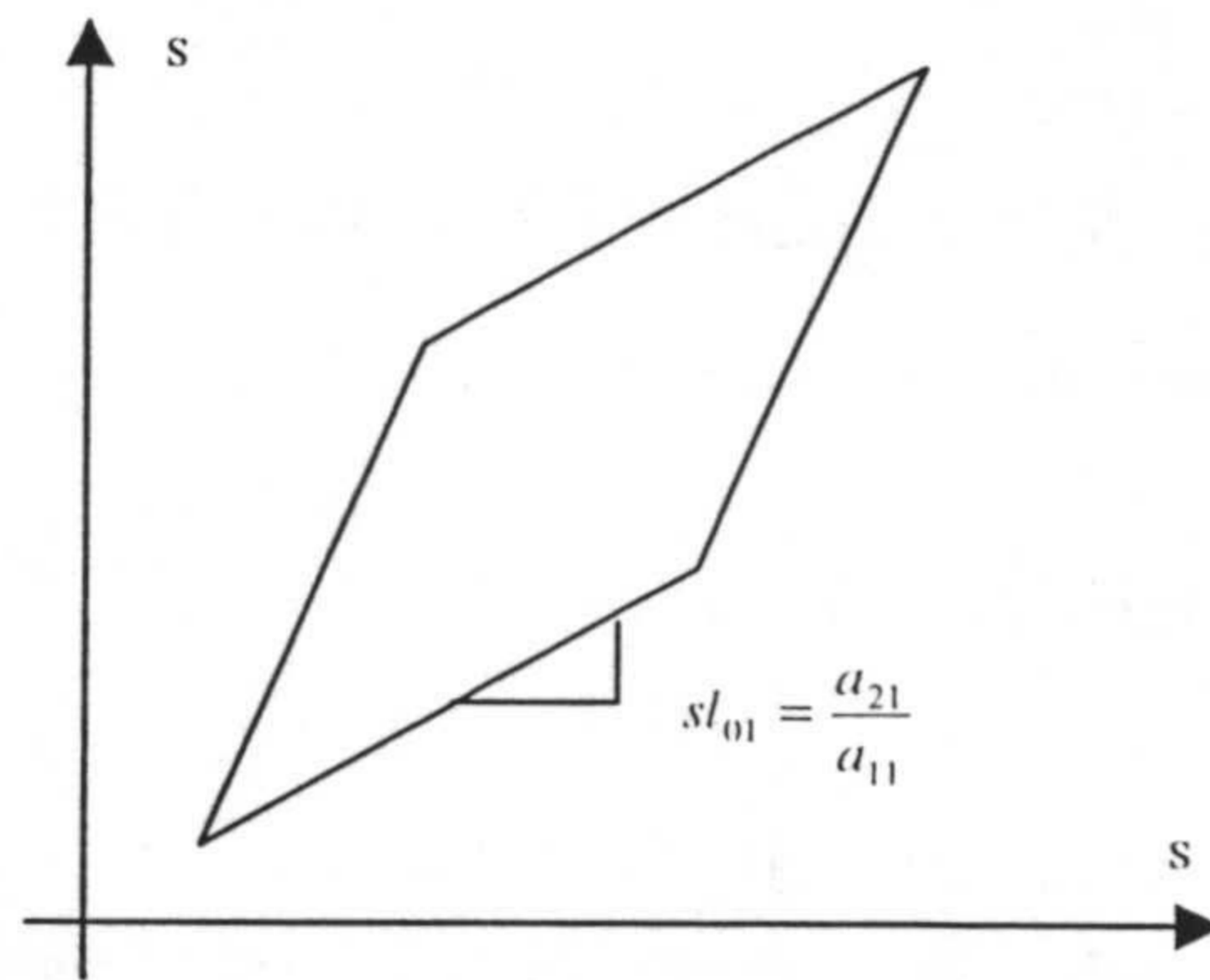


Figura 5.9. Pendientes del hiperparalelepípedo de observaciones.

Considerando 360 clases ($n_c=180$) cada clase abarcará un ángulo (*ángulo base*):

$$\text{ángulo base: } \alpha_c = \frac{\pi}{n_c} = \frac{\pi}{180} = 0.0175 \text{ Rad} \quad (5.19)$$

A partir de 5.17 podemos calcular los ángulos, α_{M1} , α_{M2} , en radianes, que forman las pendientes, $s_{l_{01}}$ y $s_{l_{02}}$, con el eje positivo de abscisas y a partir de estos ángulos determinar teóricamente las clases a las que pertenecen las direcciones de las distribuciones (clases de máxima frecuencia) para una matriz $\mathbf{A}=(a_{ij})$ dada. En efecto, los ángulos α_{M1} y α_{M2} en función de los elementos de la matriz \mathbf{A} son:

$$\begin{aligned} \alpha_{M1} &= \text{arctg}\left(\frac{a_{21}}{a_{11}}\right) \\ \alpha_{M2} &= \text{arctg}\left(\frac{a_{22}}{a_{12}}\right) \end{aligned} \quad (5.20)$$

Los vectores ganadores estarán en las clases, c_{M1} y c_{M2} , dadas por:

$$\begin{aligned} c_{M1} &= \text{ent}\left[\frac{\alpha_{M1}}{\text{ángulo base}}\right] + 1 \\ c_{M2} &= \text{ent}\left[\frac{\alpha_{M2}}{\text{ángulo base}}\right] + 1 \end{aligned} \quad (5.21)$$

Donde $\text{ent}[-]$ representa el cociente de la división entera.

Aplicando las ecuaciones (5.20 y 5.21) se obtienen, para $n_c=180$, los valores que se indican en las columnas 7 y 8 de la Tabla 5.3.

Tabla 5.3. Resultados teóricos (con $n_c=180$).

Matriz A	A_d	α_{M1} (Rad)	α_{M2} (Rad)	$\frac{\alpha_{Mn}}{\text{angbase}}$		Clases ganadoras ($n_c=180$)	
$A_1 = \begin{pmatrix} 1 & 0.5 \\ 0.5 & 1 \end{pmatrix}$	$A_{d1} = \begin{pmatrix} 1 & 0.5 \\ 0.5 & 1 \end{pmatrix}$	0.46	1.10	26.49	63.27	27	64
$A_2 = \begin{pmatrix} 0.8 & 0.1 \\ 0.2 & 0.9 \end{pmatrix}$	$A_{d2} = \begin{pmatrix} 1 & 0.11 \\ 0.25 & 1 \end{pmatrix}$	0.25	1.46	13.99	83.44	14	84
$A_3 = \begin{pmatrix} 0.8 & 0.4 \\ -0.5 & 0.9 \end{pmatrix}$	$A_{d3} = \begin{pmatrix} 1 & 0.44 \\ -0.62 & 1 \end{pmatrix}$	2.58	1.15	66.07	147.59	67	148
$A_4 = \begin{pmatrix} 0.3 & -0.1 \\ -0.4 & -1 \end{pmatrix}$	$A_{d4} = \begin{pmatrix} 1 & 0.1 \\ -1.33 & 1 \end{pmatrix}$	2.21	1.47	126.53	85.06	127	85
$A_5 = \begin{pmatrix} -0.7 & 0.1 \\ 0.1 & -1 \end{pmatrix}$	$A_{d5} = \begin{pmatrix} 1 & -0.1 \\ -0.14 & 1 \end{pmatrix}$	2.99	1.67	171.41	95.45	172	96
$A_6 = \begin{pmatrix} -1 & -0.1 \\ -0.8 & 1 \end{pmatrix}$	$A_{d6} = \begin{pmatrix} 1 & -0.1 \\ 0.8 & 1 \end{pmatrix}$	0.68	1.67	38.56	95.45	39	96
$A_7 = \begin{pmatrix} 0.6 & 0.7 \\ -0.3 & 1.4 \end{pmatrix}$	$A_{d7} = \begin{pmatrix} 1 & 0.5 \\ -0.5 & 1 \end{pmatrix}$	2.68	1.11	153.03	63.27	154	64
$A_8 = \begin{pmatrix} 0.8 & -0.88 \\ 0.77 & 0.9 \end{pmatrix}$	$A_{d8} = \begin{pmatrix} 1 & -0.98 \\ 0.97 & 1 \end{pmatrix}$	0.77	2.35	45.01	135.24	45	135

La segunda columna de la Tabla 5.3 muestra las 8 matrices de mezcla utilizadas, en su forma *canónica*, es decir, con su diagonal principal a unos (ver Sec.2.5.2.1).

A continuación se presentan los resultados obtenidos. En primer lugar (Sección 5.5.1) se mostrarán los resultados obtenidos con fuentes sintéticas, y posteriormente con señales de voz reales (Sección 5.5.2).

5.4.1 Resultados con señales sintéticas

Para validar el método propuesto se han generado, con el entorno de simulación descrito en el Capítulo 3 (Sec.3.3.1), señales aleatorias que siguen distribuciones de probabilidad laplaciana (Sec.A1.3.4) y gamma bilateral (Sec.A1.3.3), con desviaciones estándar dentro

del rango de señales de audio. En la Tabla 5.4 y en la Figura 5.10 se muestran distintos resultados obtenidos con señales aleatorias con densidad de probabilidad laplaciana.

Tabla 5.4. Señales sintéticas laplacianas.

Desviación estándar		M	n_c	Clases ganadoras	A_d	W_{di}	$\bar{\epsilon}_a$
15	18	100.000	360	127 y 62	$\begin{pmatrix} 1 & 0.5 \\ 0.5 & 1 \end{pmatrix}$	$\begin{pmatrix} 1 & 0.5151 \\ 0.5949 & 1 \end{pmatrix}$	0.028
15	18	100.000	360	33 y 167	$\begin{pmatrix} 1 & 0.11 \\ 0.25 & 1 \end{pmatrix}$	$\begin{pmatrix} 1 & 0.1184 \\ 0.2916 & 1 \end{pmatrix}$	0.013
15	18	300.000	360	133 y 293	$\begin{pmatrix} 1 & 0.44 \\ -0.62 & 1 \end{pmatrix}$	$\begin{pmatrix} 1 & 0.44 \\ -0.662 & 1 \end{pmatrix}$	0.021
15	16	100.000	180	127 y 85	$\begin{pmatrix} 1 & 0.1 \\ -1.33 & 1 \end{pmatrix}$	$\begin{pmatrix} 1 & 0.09 \\ -1.4 & 1 \end{pmatrix}$	0.019
12	14	150.000	180	171 y 97	$\begin{pmatrix} 1 & -0.1 \\ -0.14 & 1 \end{pmatrix}$	$\begin{pmatrix} 1 & -0.1138 \\ -0.1676 & 1 \end{pmatrix}$	0.010
15	16	100.000	360	79 y 189	$\begin{pmatrix} 1 & -0.1 \\ 0.8 & 1 \end{pmatrix}$	$\begin{pmatrix} 1 & -0.0744 \\ 0.8169 & 1 \end{pmatrix}$	0.011
20	30	50.000	360	299 y 131	$\begin{pmatrix} 1 & 0.5 \\ -0.5 & 1 \end{pmatrix}$	$\begin{pmatrix} 1 & 0.49 \\ -0.49 & 1 \end{pmatrix}$	0.005

Las columnas de la Tabla 5.4 contienen las desviaciones estándar de las dos señales mezcladas, el número de muestras utilizado (M), el número de clases considerado (n_c), las clases ganadoras obtenidas por el procedimiento, la forma canónica de la matriz de mezcla usada, la matriz de mezcla obtenida y la media del error absoluto obtenido entre los elementos de la matriz de mezcla estimada y la original ($\bar{\epsilon}_a$) (Séc.2.5.2.1, (2.106)). Puede observarse que en todos los casos los resultados son satisfactorios (se obtiene un error menor de 3 centésimas).

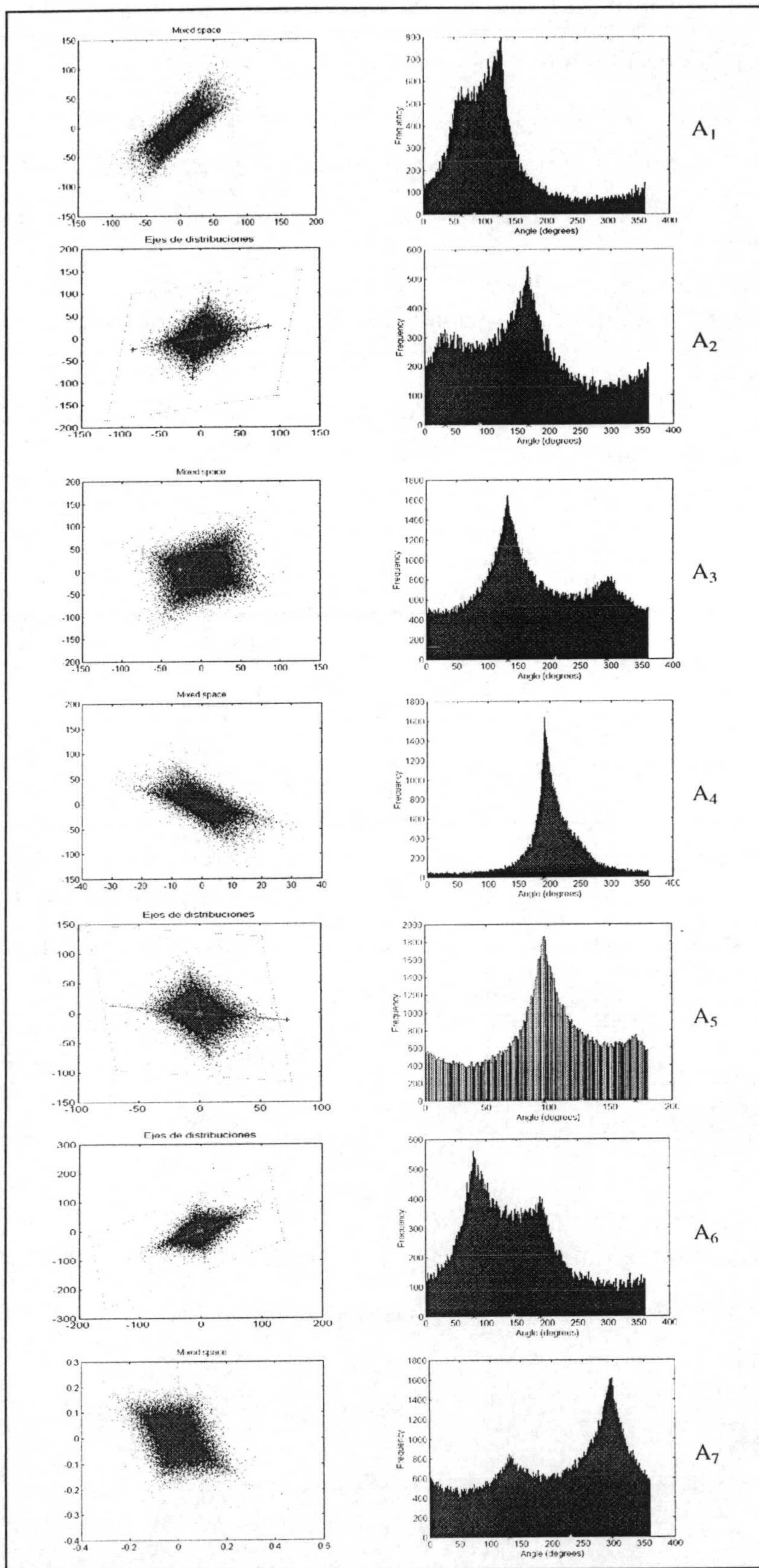


Figura 5. 10. Señales sintéticas Laplacianas: en cada fila se incluye: espacio de mezclas, histograma de frecuencias y matriz de mezcla.

La Figura 5.9 muestra, para todos los casos considerados en la Tabla 5.4 el espacio de observaciones y la distribución angular de frecuencias.

En la Tabla 5.5 y en la Figura 5.11 se muestran los resultados obtenidos con señales gamma bilateral al realizar el mismo proceso que con las señales laplacianas.

Tabla 5.5. Señales sintéticas gamma bilateral.

Desv. estándar		NM	n_c	Clases ganadoras	A_d	W_{di}	\mathcal{E}_a
15	18	50.000	180	27 y 64	$\begin{pmatrix} 1 & 0.5 \\ 0.5 & 1 \end{pmatrix}$	$\begin{pmatrix} 1 & 0.49 \\ 0.49 & 1 \end{pmatrix}$	0.0006
15	18	50.000	180	15 y 84	$\begin{pmatrix} 1 & 0.11 \\ 0.25 & 1 \end{pmatrix}$	$\begin{pmatrix} 1 & 0.11 \\ 0.25 & 1 \end{pmatrix}$	0.002
15	18	50.000	180	67 y 148	$\begin{pmatrix} 1 & 0.44 \\ -0.62 & 1 \end{pmatrix}$	$\begin{pmatrix} 1 & 0.43 \\ -0.63 & 1 \end{pmatrix}$	0.004
15	16	100.000	180	127 y 85	$\begin{pmatrix} 1 & 0.1 \\ -1.33 & 1 \end{pmatrix}$	$\begin{pmatrix} 1 & 0.09 \\ -1.34 & 1 \end{pmatrix}$	0.0051
12	14	30.000	180	173 y 96	$\begin{pmatrix} 1 & -0.1 \\ -0.14 & 1 \end{pmatrix}$	$\begin{pmatrix} 1 & -0.09 \\ -0.13 & 1 \end{pmatrix}$	0.0070
15	16	40.000	180	39 y 97	$\begin{pmatrix} 1 & -0.1 \\ 0.8 & 1 \end{pmatrix}$	$\begin{pmatrix} 1 & -0.11 \\ 0.79 & 1 \end{pmatrix}$	0.0086
20	30	50.000	180	64 y 154	$\begin{pmatrix} 1 & 0.5 \\ -0.5 & 1 \end{pmatrix}$	$\begin{pmatrix} 1 & 0.49 \\ -0.49 & 1 \end{pmatrix}$	0.0012

Al ser la distribución gamma bilateral más “estilizada”, (Figura A1.4), para iguales valores de la desviación estándar se obtienen resultados mejores: error medio menor de 87 diezmilésimas.

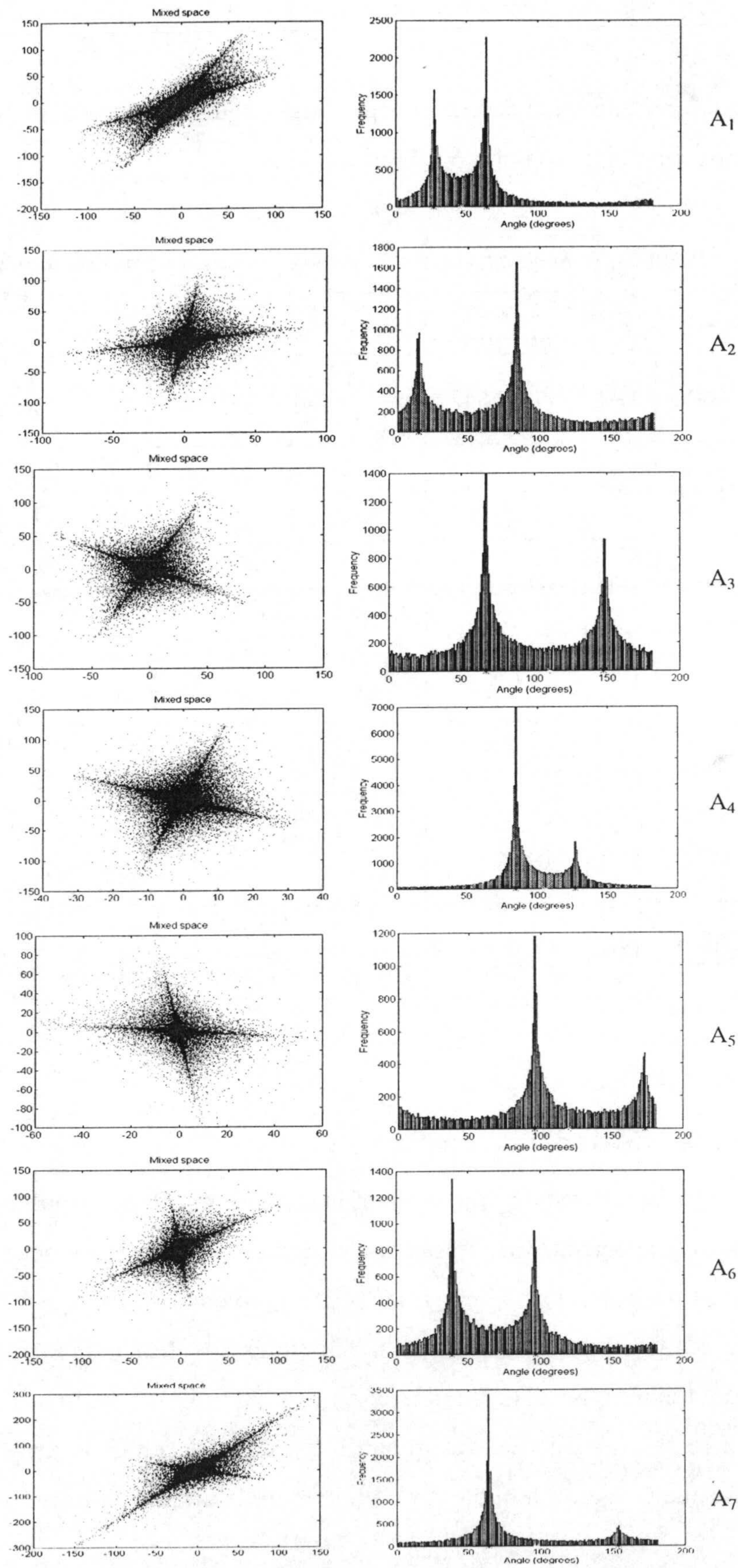


Figura 5. 11. Señales sintéticas gamma bilateral: en cada fila se incluye: espacio de mezclas, histograma de frecuencias y matriz de mezcla.

También se han probado mezclas de señales laplacianas con gamma bilaterales. Los resultados se muestran en la Tabla 5.6 y Figura 5.12.

Tabla 5. 6. *Señales sintéticas Gamma bilateral y Laplacianas.*

Desv. estándar		NM	n_c	Clases ganadoras	A_d	W_{di}	\mathcal{E}_a
15	18	100.000	180	27 y 64	$\begin{pmatrix} 1 & 0.5 \\ 0.5 & 1 \end{pmatrix}$	$\begin{pmatrix} 1 & 0.49 \\ 0.49 & 1 \end{pmatrix}$	0.0011
15	18	50.000	180	13 y 84	$\begin{pmatrix} 1 & 0.11 \\ 0.25 & 1 \end{pmatrix}$	$\begin{pmatrix} 1 & 0.11 \\ 0.22 & 1 \end{pmatrix}$	0.0015
15	18	50.000	180	67 y 147	$\begin{pmatrix} 1 & 0.44 \\ -0.62 & 1 \end{pmatrix}$	$\begin{pmatrix} 1 & 0.43 \\ -0.63 & 1 \end{pmatrix}$	0.0225
15	16	50.000	180	127 y 85	$\begin{pmatrix} 1 & 0.1 \\ -1.33 & 1 \end{pmatrix}$	$\begin{pmatrix} 1 & 0.09 \\ -1.35 & 1 \end{pmatrix}$	0.0107
12	14	40.000	180	174 y 96	$\begin{pmatrix} 1 & -0.1 \\ -0.14 & 1 \end{pmatrix}$	$\begin{pmatrix} 1 & -0.09 \\ -0.11 & 1 \end{pmatrix}$	0.0161
15	16	100.000	180	38 y 96	$\begin{pmatrix} 1 & -0.1 \\ 0.8 & 1 \end{pmatrix}$	$\begin{pmatrix} 1 & -0.09 \\ 0.7 & 1 \end{pmatrix}$	0.0179
20	30	100.000	180	64 y 160	$\begin{pmatrix} 1 & 0.5 \\ -0.5 & 1 \end{pmatrix}$	$\begin{pmatrix} 1 & 0.49 \\ -0.37 & 1 \end{pmatrix}$	0.0638

5.4.2 Resultados con señales reales.

Se han considerado 11 señales de voz, contenidas en el entorno de simulación descrito en el Capítulo 3, consistentes en frases procedentes de varios locutores grabadas directamente y otras obtenidas a partir de CDs convencionales. En la Tabla 5.7 se muestran varios parámetros calculados en las distintas señales. Estas señales se pueden aproximar a una función de distribución gamma bilateral. La Figura 5.8 muestra una comparación entre las densidades de probabilidad de una señal de voz (Juliette) y su correspondiente función gamma bilateral. Se obtienen resultados similares con el resto de las señales.

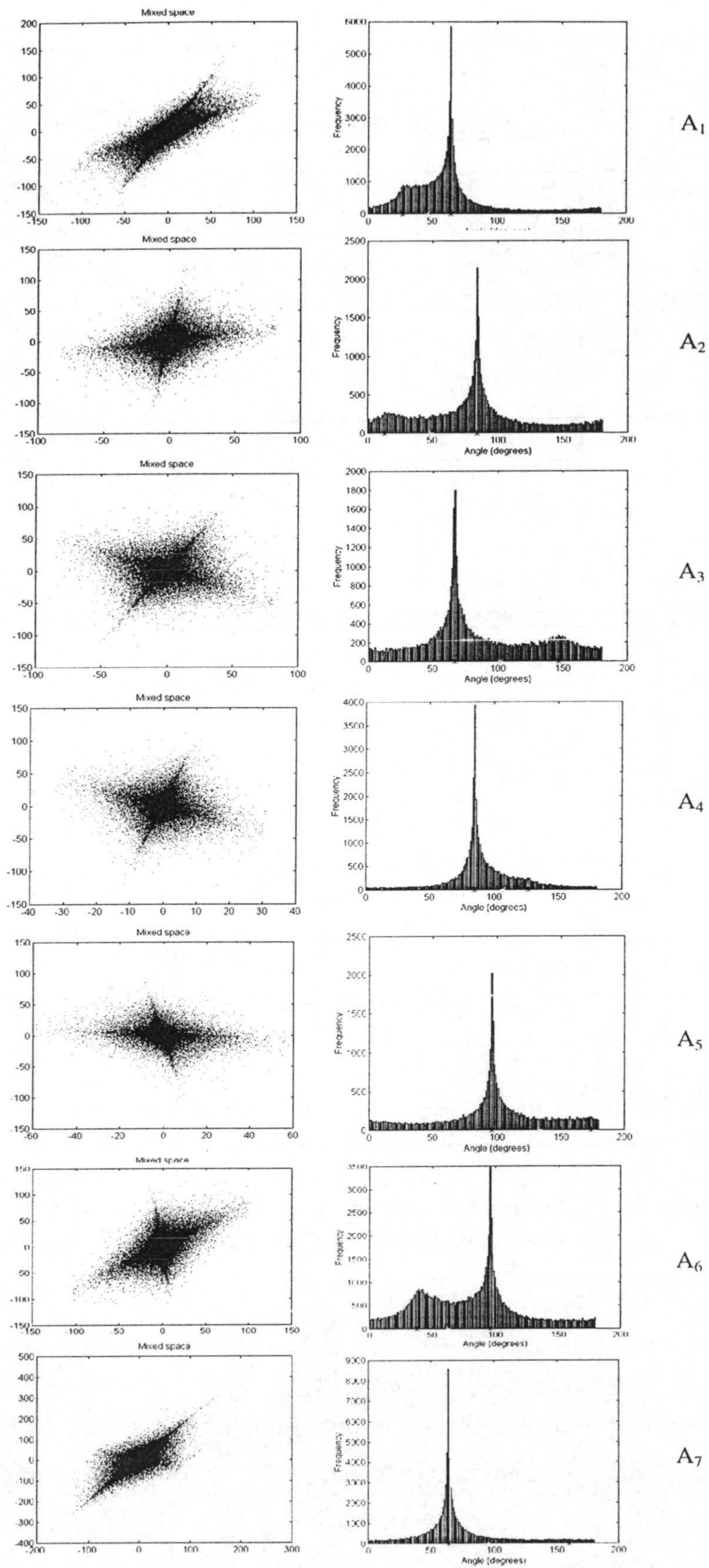


Figura 5. 12. Mezcla de las señales sintéticas laplaciana y gamma bilateral. En cada fila se incluye: espacio de mezclas, histograma de frecuencias, y matriz de mezcla.

Tabla 5. 7. Características de las señales reales escogidas

Señal	Nº Muestra	Tiempo (s)	Media	Mediana	Desviación estándar	Moda	Frec. (max)	Min (S)	Max (S)	Gamma (0)	Gamma (255)
Alberto1	163,118	7.398	126.99	128	16.19	128	7,503	5	234	6.1181E-06	6.4798E-06
Alberto2	131,705	5.973	127.00	127	12.71	127	9,206	2	226	1.1377E-06	1.0586E-06
Antonio	118,203	5.361	126.00	127	13.34	127	6,966	0	212	1.6778E-06	1.5662E-06
Begoña	143,782	6.521	127.00	127	19.68	127	6,300	8	255	1.9625E-05	1.8707E-05
Carlos	226,816	10.286	127.00	127	17.11	127	11,323	0	255	9.1118E-06	8.6283E-06
Jose L.	123,162	5.586	127.01	127	13.29	127	7,882	0	226	1.6229E-06	1.5145E-06
Juliette	66,844	3.032	127.04	127	22.52	127	5,778	2	244	3.7135E-05	3.5594E-05
Luis	100,629	5.564	126.99	127	17.86	126	4,685	16	253	1.1651E-05	1.1056E-05
Mancia	117,705	5.338	127.01	127	20.88	127	4,551	0	251	2.6291E-05	2.5124E-05
Orson	224,360	10.175	127.02	127	11.76	127	39,736	22	255	5.9075E-07	5.4668E-07
Pulp Fiction	117,376	5.323	127.00	127	21.11	127	7,812	8	254	2.7665E-05	2.6449E-05

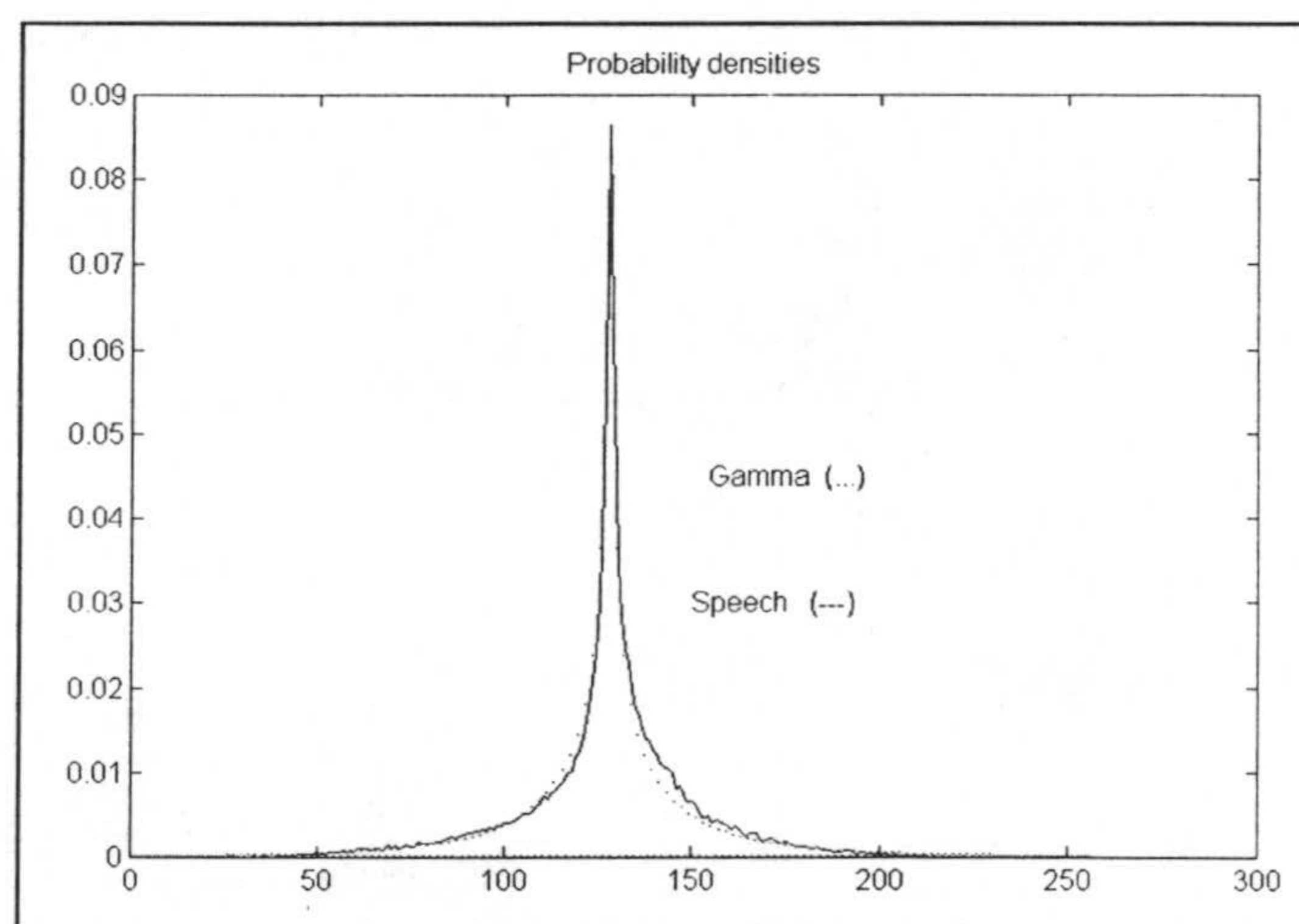


Figura 5. 13. Densidades de probabilidad de una señal real (Jul3) y una función gamma bilateral con las mismas μ y σ .

Las señales de voz usadas se digitalizaron con una tarjeta de sonido Sound Blaster y fueron captadas con el programa Wave Studio. La conversión se llevó a cabo con una precisión de 8 bits a una frecuencia de muestreo de 22.050 Hz (calidad "mono-radio", Tabla 5.8).

Tabla 5. 8. Calidad de señales de audio

	Formato	Nº de bits	Frecuencia de muestreo (Hz)	Período de muestreo (s)
Radio	PCM	8 mono	22050	$5.53 \cdot 10^{-5}$
Teléfono	PCM	8 mono	11025	$9.07 \cdot 10^{-5}$
CD	PCM	16 estéreo	44100	$2.27 \cdot 10^{-5}$

Los algoritmos propuestos en la Sec.5.3 se han probado con las señales descritas en la Tabla 5.7. y con las matrices de la Tabla 5.2, obteniéndose resultados satisfactorios en cada caso. Se han utilizado mezclas de voz (voz + música) y (música + música) y en todos los casos el tiempo de las señales de mezcla necesario para hacer satisfactoriamente la separación ha sido menor de 2.25 segundos.

Todas las pruebas y resultados se han obtenido con el simulador descrito en el Capítulo 3.

En la Tabla 5.9 se muestran algunos de los resultados experimentales. En la última columna se da la media del error absoluto, $\bar{\epsilon}_a$, obtenido entre la matriz de mezcla estimada y la original. En todos los casos es menos de una centésima.

Tabla 5. 9. Resultados experimentales.

	Señales	NM	n_c	Clases ganadoras	A_d	W_{di}	$\bar{\epsilon}_a$
Ejemplo 1	Alberto Begoña	50.000	180	27 y 63	$A_{d1} = \begin{pmatrix} 1 & 0.5 \\ 0.5 & 1 \end{pmatrix}$	$\begin{pmatrix} 1 & 0.52 \\ 0.49 & 1 \end{pmatrix}$	0.008
Ejemplo 2	Beethoven Mozart	50.000	180	15 y 84	$A_{d2} = \begin{pmatrix} 1 & 0.11 \\ 0.25 & 1 \end{pmatrix}$	$\begin{pmatrix} 1 & 0.11 \\ 0.25 & 1 \end{pmatrix}$	0.00
Ejemplo 3	Orson Wells Pulp Fiction	50.000	180	67 y 148	$A_{d3} = \begin{pmatrix} 1 & 0.44 \\ -0.62 & 1 \end{pmatrix}$	$\begin{pmatrix} 1 & 0.44 \\ -0.63 & 1 \end{pmatrix}$	0.003
Ejemplo 4	Luis Begoña	40.000	180	126 y 85	$A_{d4} = \begin{pmatrix} 1 & 0.1 \\ -1.33 & 1 \end{pmatrix}$	$\begin{pmatrix} 1 & 0.09 \\ -1.33 & 1 \end{pmatrix}$	0.003
Ejemplo 5	Beethoven Antonio	40.000	180	172 y 96	$A_{d5} = \begin{pmatrix} 1 & -0.1 \\ -0.14 & 1 \end{pmatrix}$	$\begin{pmatrix} 1 & -0.10 \\ -0.14 & 1 \end{pmatrix}$	0.00
Ejemplo 6	Carlos Manolo	30.000	180	39 y 96	$A_{d6} = \begin{pmatrix} 1 & -0.1 \\ 0.8 & 1 \end{pmatrix}$	$\begin{pmatrix} 1 & -0.10 \\ 0.80 & 1 \end{pmatrix}$	0.00

En las Figuras 5.14 a 5.19 se muestran para los 6 ejemplos de la Tabla 5.9: a) espacios fuente, b) espacios de observaciones, c) histogramas de frecuencias de los ángulos, d) derivadas de las frecuencias, e) evolución en el tiempo de las señales originales y f) de las

recuperadas, g) error cuadrático medio (RMS), h) índice de Amari, i) diafonía, y j) error absoluto. Estas medidas de prestaciones se han obtenido como se indicó en la Sec.2.5.

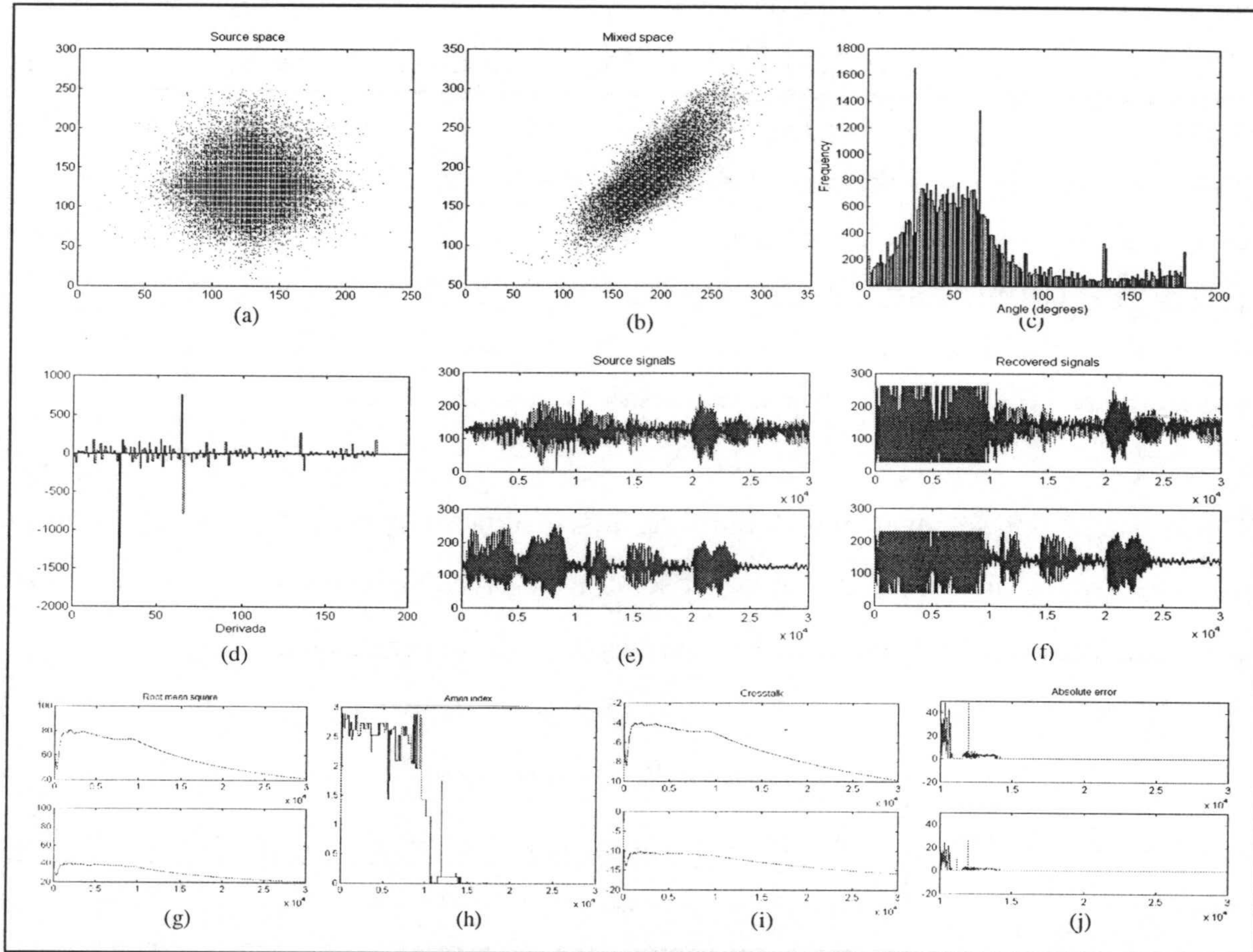


Figura 5. 14. Ejemplo 1.

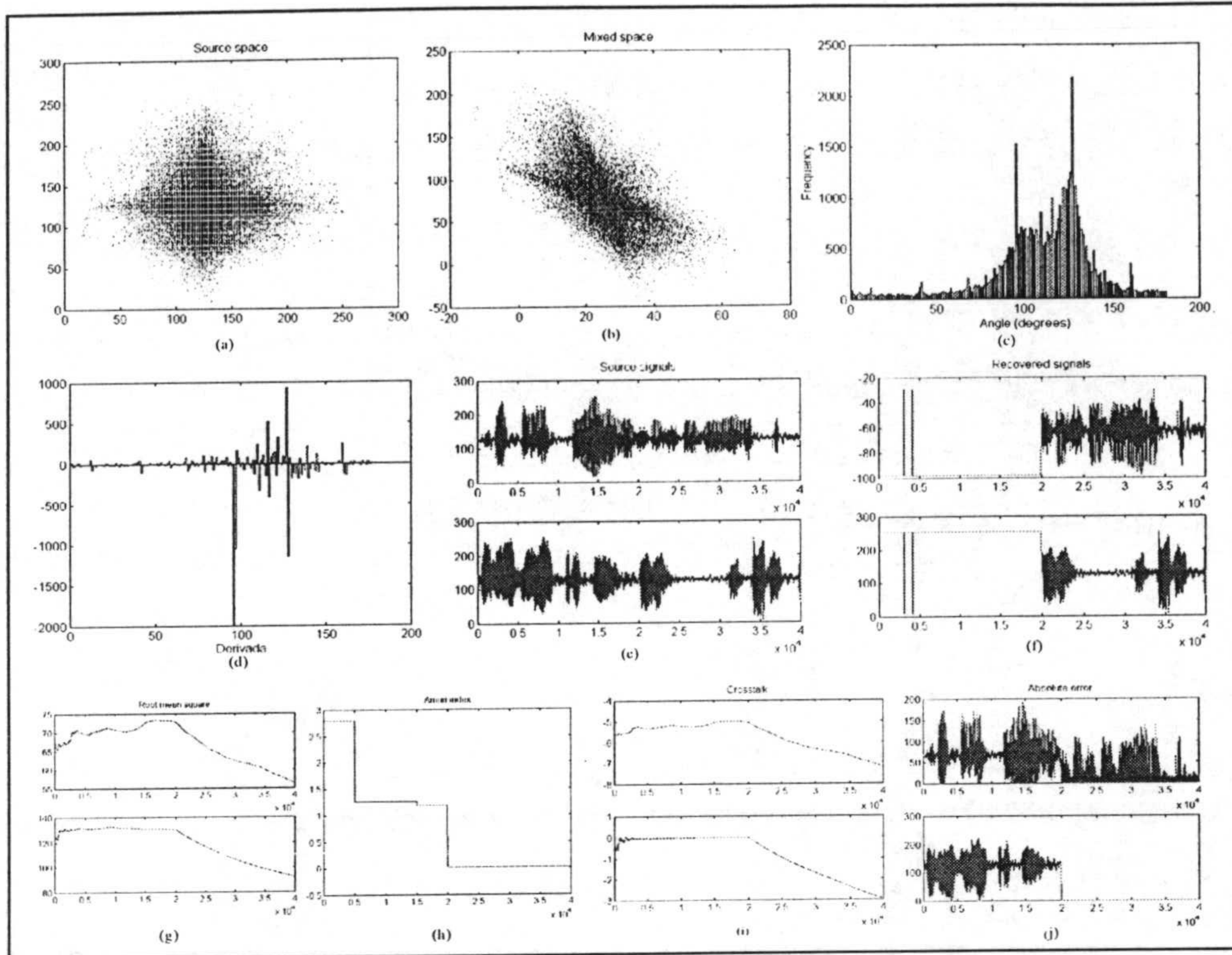


Figura 5. 15. Ejemplo 2

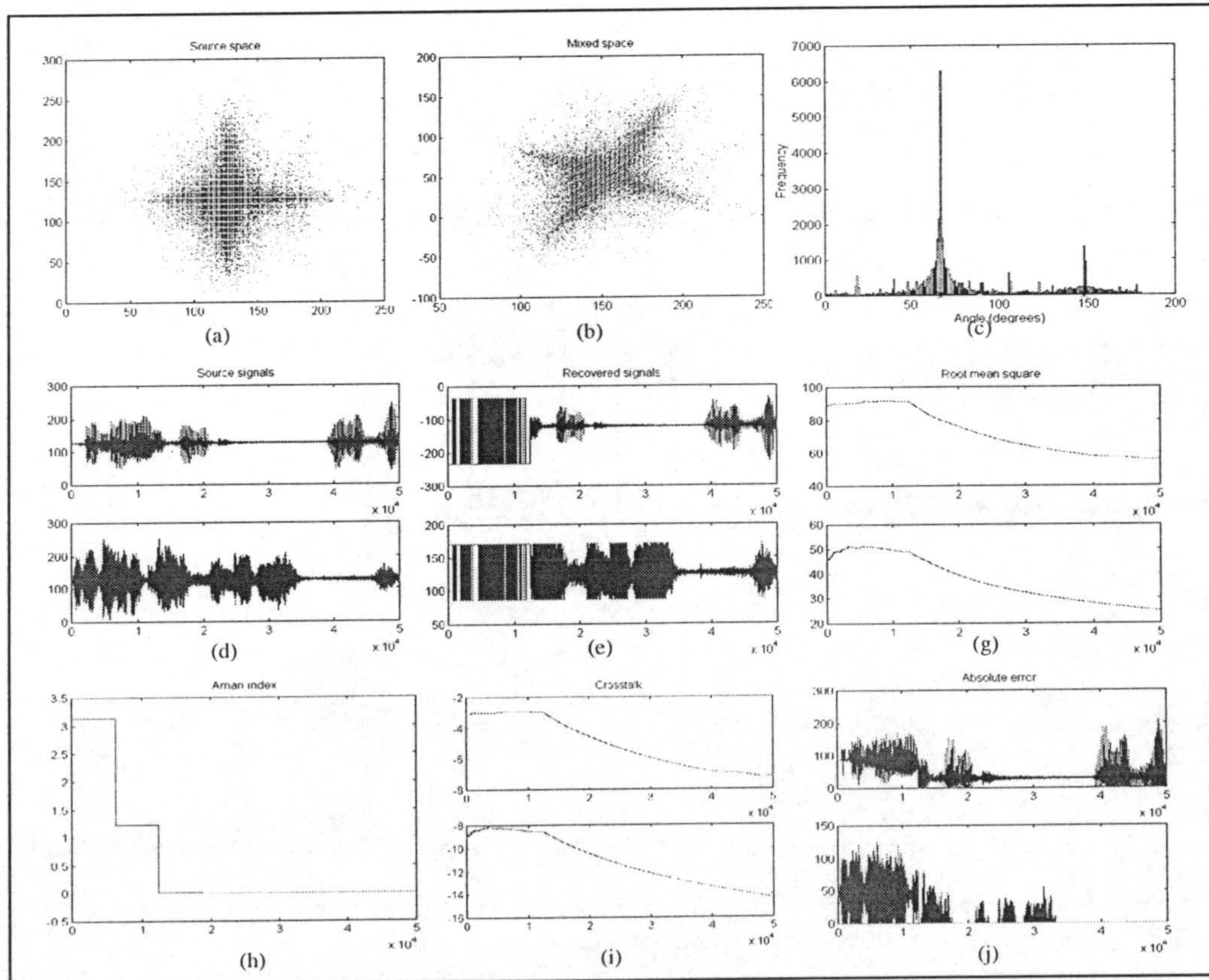


Figura 5. 16. Ejemplo 3.

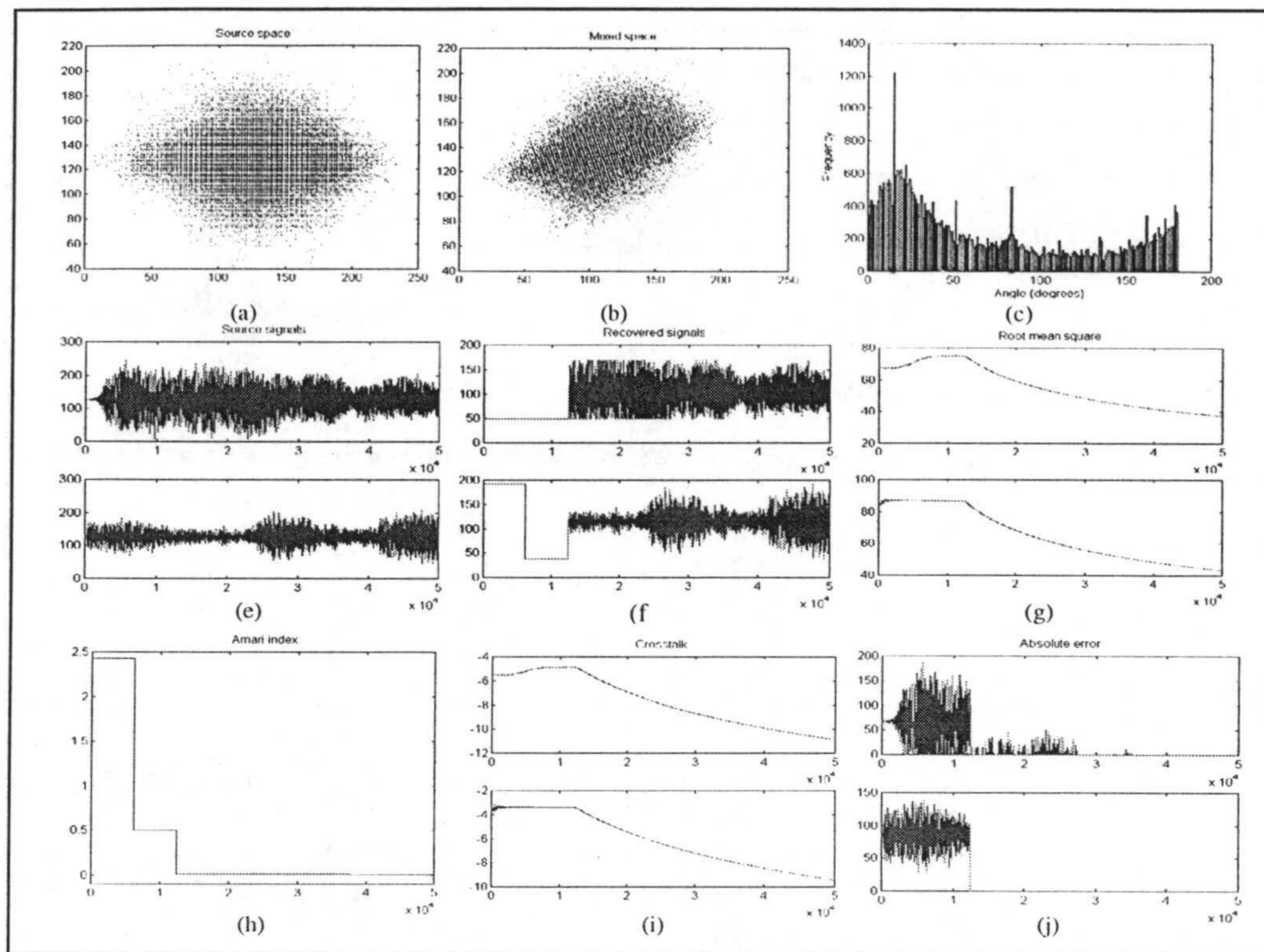


Figura 5.17. Ejemplo 5.

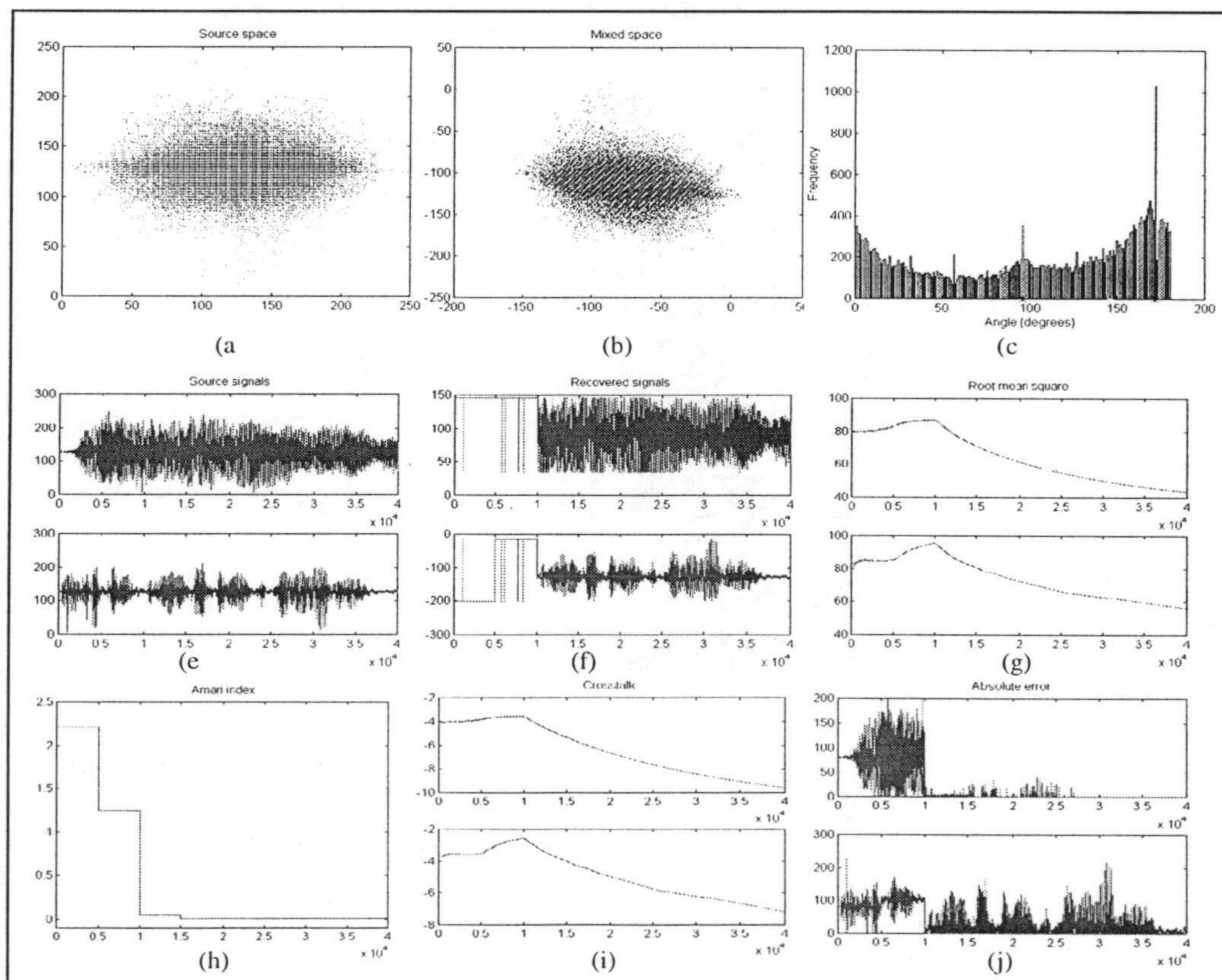


Figura 5.18. Ejemplo 5.

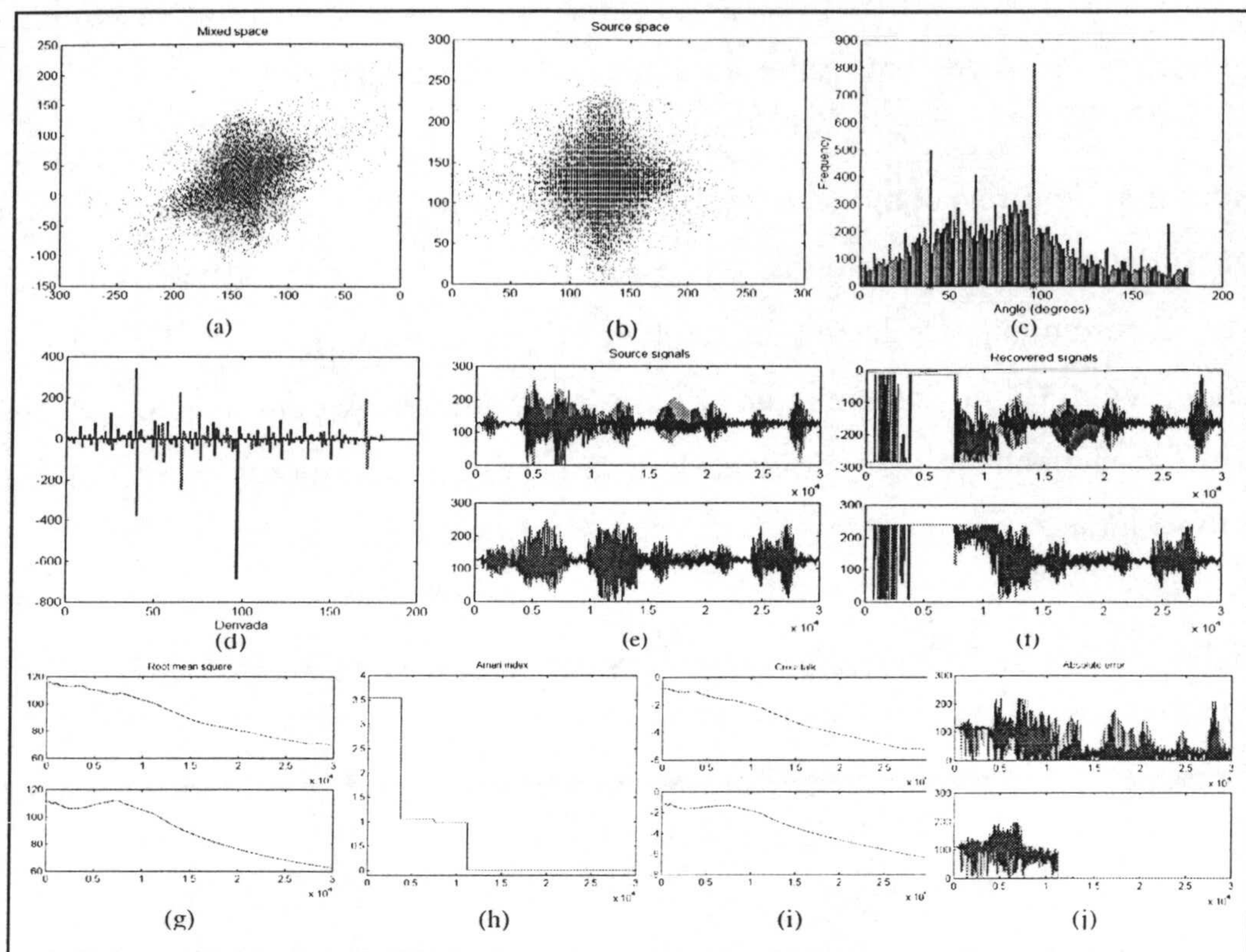


Figura 5.19. Ejemplo 6.

5.5 DEPENDENCIA DE LA ESTIMACIÓN DE LA MATRIZ DE MEZCLA CON EL NÚMERO DE CLASES CONSIDERADAS.

El procedimiento de estimación de la matriz de mezcla propuesto en este capítulo (Sec.5.3.1) puede resumirse en:

- 1) determinar las dos clases donde la frecuencia angular de las observaciones es mayor (c_{M1} , c_{M2}) y
- 2) determinar los vectores representantes de dichas clases, w_{M1} , w_{M2} , ya que estos serán las columnas de la matriz W que estima a A .

De acuerdo con lo anterior pueden producirse dos tipos de errores en la estimación de A que se reducirán a lo largo del proceso iterativo, si este es convergente. El primer tipo se refiere al error debido a la obtención de clases ganadoras distintas de las teóricas (es

decir, distintas a las correspondientes columnas de \mathbf{A}). El segundo está ligado a la determinación de los vectores representantes de las clases ganadoras.

El primer tipo de error, como se demostrará a continuación, puede reducirse aumentando el número de clases, n_c , consideradas. La reducción del error en la determinación de los vectores representantes de las clases también se produce en el caso del algoritmo “fuera de línea” (Sec.5.3.3) al incrementar el número de clases. A continuación se efectúa un análisis de la dependencia del error en la estimación de los valores a_{ij} con el número de clases consideradas.

Para dicho análisis se considera una matriz \mathbf{A} canónica y por tanto con la siguiente forma:

$$A_d = \begin{pmatrix} 1 & a_{d12} \\ a_{d21} & 1 \end{pmatrix} \quad (5.25)$$

con: $a_{d21} = \frac{a_{21}}{a_{11}}, \quad a_{d12} = \frac{a_{12}}{a_{22}}$

Para llegar a una relación entre los valores de los coeficientes de \mathbf{A} y el número de clases se parte de la relación entre los coeficientes de \mathbf{A} y las clases correspondientes a los máximos de la distribución de frecuencias, α_{Mi} , que según (5.20) viene dada por:

$$\begin{aligned} a_{d21} &= \operatorname{tg}(\alpha_{M1}) \\ a_{d12} &= \frac{1}{\operatorname{tg}(\alpha_{M2})} \end{aligned} \quad (5.26)$$

Derivando la expresión (5.26) con respecto a α se obtiene:

$$\begin{aligned} \frac{da_{d21}}{d\alpha_{M1}} &= \frac{1}{\cos^2(\alpha_{M1})} \\ \frac{da_{d12}}{d\alpha_{M2}} &= -\frac{1}{\operatorname{sen}^2(\alpha_{M2})} \end{aligned} \quad (5.27)$$

Aproximando la derivada por incrementos se tiene que:

$$\begin{aligned}\Delta a_{d21} &= \frac{1}{\cos^2(\alpha_{M1})} \cdot \Delta\alpha_{M1} \\ \Delta a_{d12} &= -\frac{1}{\sin^2(\alpha_{M2})} \cdot \Delta\alpha_{M2}\end{aligned}\tag{5.28}$$

Las expresiones (5.28) proporcionan la desviación de los valores de los elementos de A_d en función del error en la determinación de los ángulos α_{M1} y α_{M2} .

Como $a_{d21} = \text{tg}(\alpha_{M1})$ y $a_{d12} = \frac{1}{\text{tg}(\alpha_{M2})}$ se puede obtener la expresión (5.28) en función de los elementos de A :

$$\begin{aligned}\Delta a_{d21} &= (1 + a_{d21}^2) \cdot \Delta\alpha_{M1} \\ \Delta a_{d12} &= -(1 + a_{d12}^2) \cdot \Delta\alpha_{M2}\end{aligned}\tag{5.29}$$

Suponiendo que se produce un error en la determinación de la clase para la que la frecuencia es máxima, de forma que se obtiene una clase contigua, se verificará (5.19):

$$\Delta\alpha = \frac{\pi}{n_c}\tag{5.30}$$

sustituyendo en (5.29):

$$\begin{aligned}\Delta a_{d21} &= (1 + a_{d21}^2) \cdot \frac{\pi}{n_c} \\ \Delta a_{d12} &= -(1 + a_{d12}^2) \cdot \frac{\pi}{n_c}\end{aligned}\tag{5.31}$$

De la expresión (5.31) se deduce que si aumenta el número de clases, n_c , disminuye el error cometido en la estimación de los coeficientes de A (es decir, Δa_{d21} y Δa_{d12} disminuyen).

En la Figura 5.20 se muestra la variación de los coeficientes, a_{d21} y a_{d12} , para distinto número de clases ($n_c = 90, 180$ y 360) en función de los valores iniciales de a_{d21} y a_{d12} .

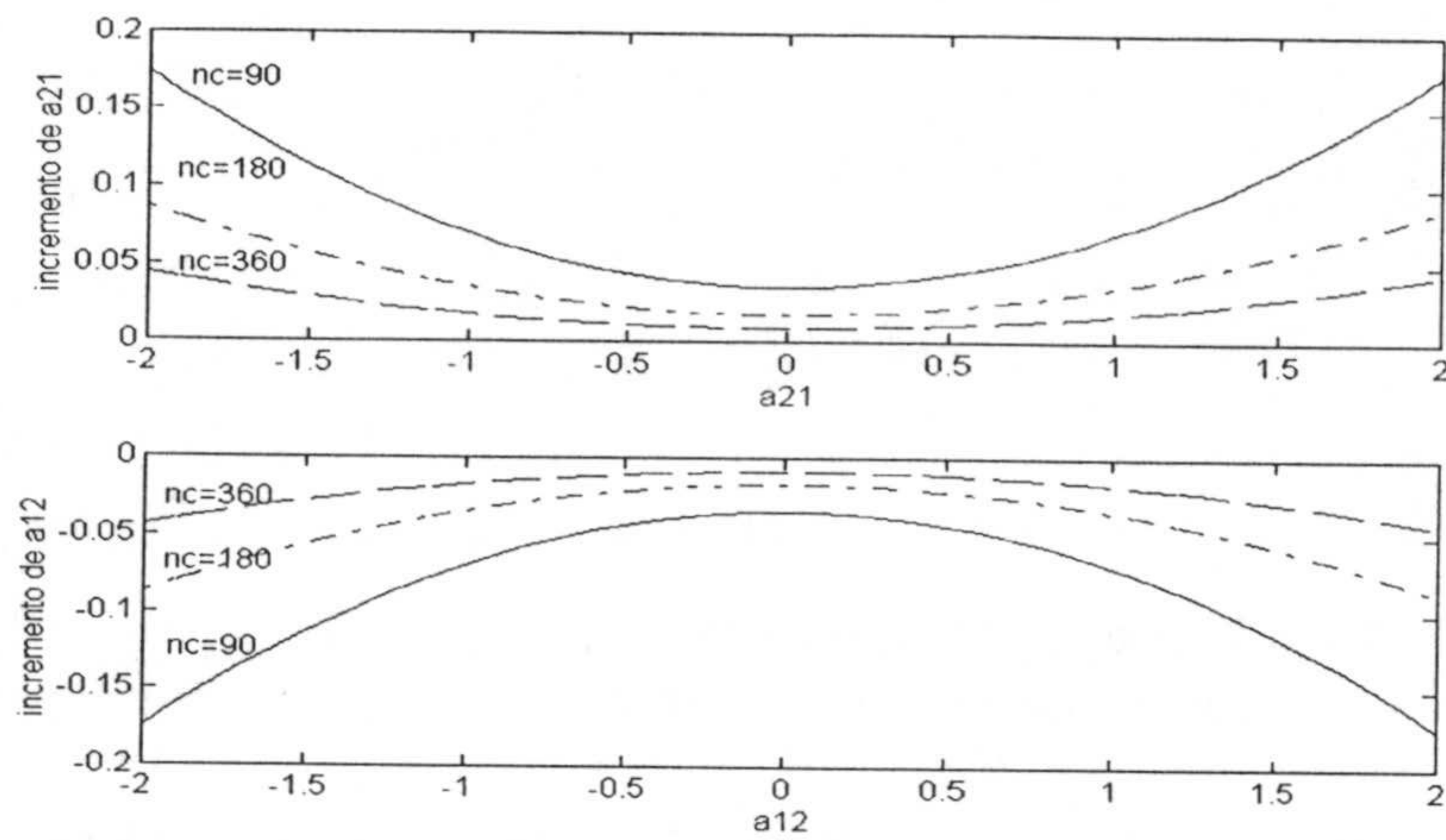


Figura 5.20 Variación de los coeficientes, a_{d21} y a_{d12} , para distinto número de clases ($n_c = 90, 180$ y 360).

Concretamente, en el caso de considerar $n_c = 180$ clases y las matrices de mezcla A_1, A_2, \dots, A_8 se obtiene la Figura 5.21 se indican con asterisco (*) los puntos correspondientes a los elementos de las matrices A_1, A_2, \dots, A_8 consideradas en este Capítulo. Se observa que el error mínimo aproximadamente es $\varepsilon_L = \pm 0.02$, y consideramos que es un *valor límite del error* medio de estimación de la matriz A . En otras palabras, convenimos que la separación es correcta cuando $\bar{\varepsilon} \leq \varepsilon_L$.

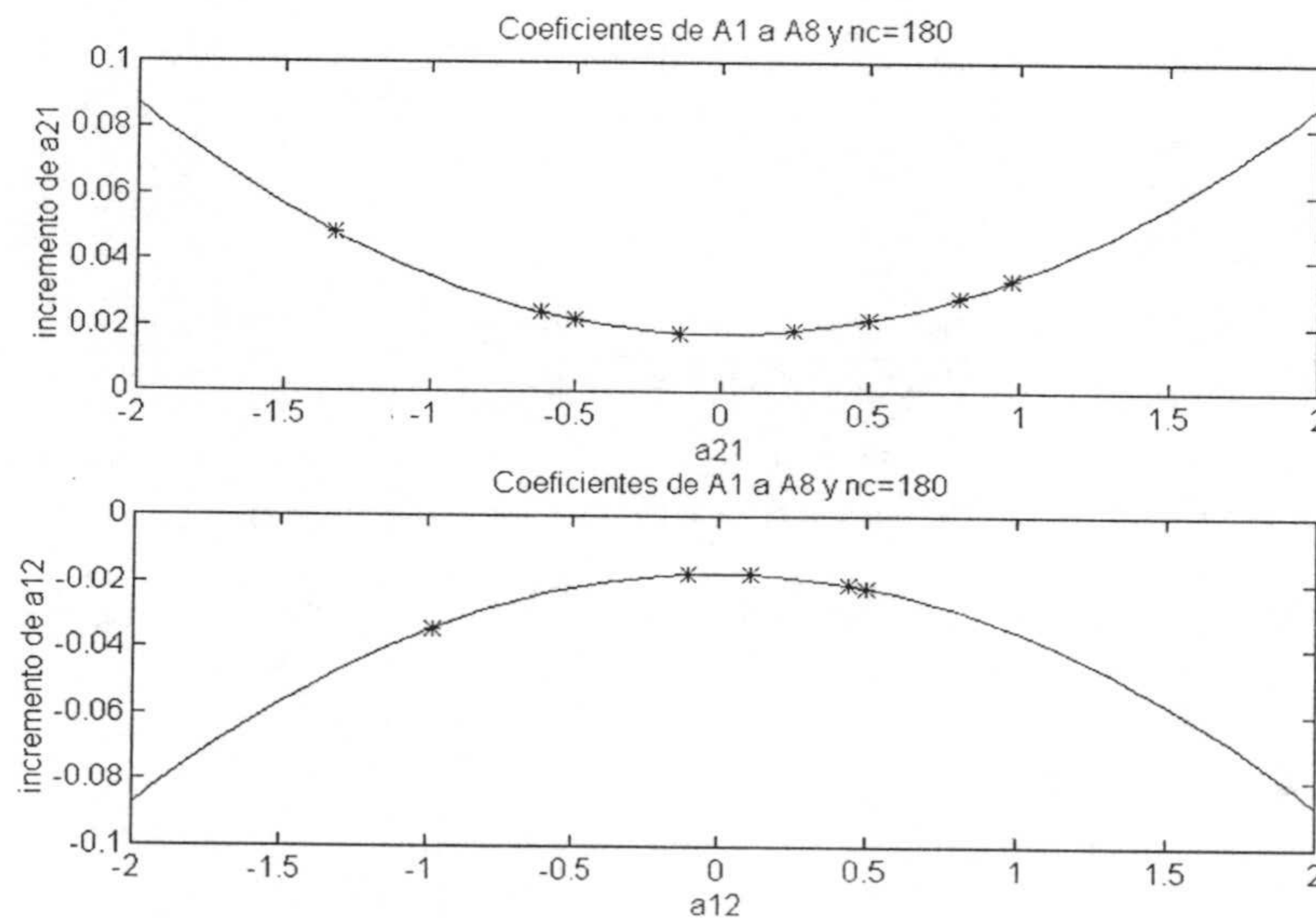


Figura 5.21 Variación de los coeficientes, a_{d21} y a_{d12} , de la matrices A_1, A_2, \dots, A_8 para $n_c = 180$ (cada punto corresponde a una matriz).

Por lo que respecta al posible error cometido en la obtención de los vectores representantes de las clases, primero se considerará el caso del algoritmo “fuera de línea” descrito en la Sec. 5.3.1. Se recordará (Sec.5.3.3) que en este caso (Figura 5.22a) se tomaba como vector representante de la clase el vector situado en el centro de la clase. Es decir, si c_M es la clase ganadora, el ángulo correspondiente al vector representante de la clase c es:

$$\alpha_c = c_M \cdot \frac{\pi}{n_c} - \frac{1}{2} \cdot \frac{\pi}{n_c} = (c_M - 0.5) \cdot \frac{\pi}{n_c} \quad (5.32)$$

Pero tal como se muestra en la Figura 5.22a, el eje de la distribución angular de los vectores de mezcla estará en la clase ganadora, pero no tiene por qué coincidir con el centro de la clase. Es decir, se cometerá un error, ε_v , acotado por:

$$-\frac{\pi}{2n_c} \leq \varepsilon_v \leq \frac{\pi}{2n_c} \quad (5.33)$$

De (5.33) se deduce que la cota de error se reduce sin más que aumentar el número de clases, n_c .

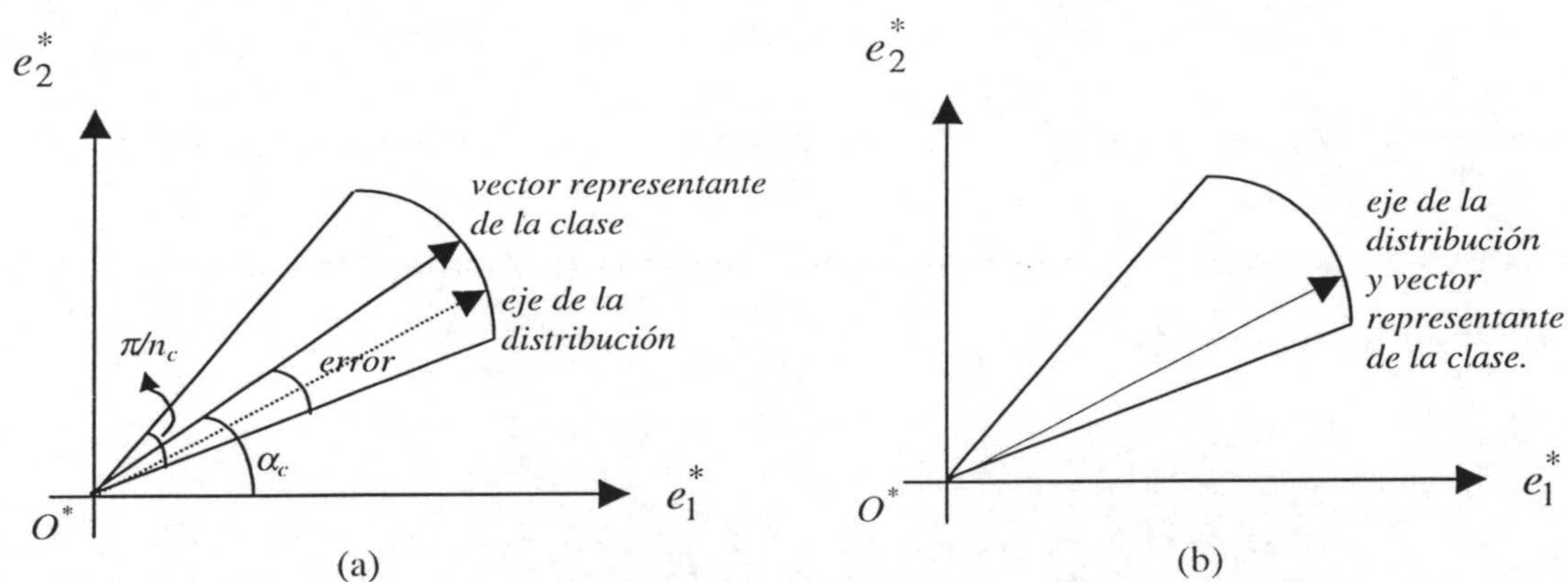


Figura 5. 22 Vector representante de una clase (el sector corresponde a la clase): (a) en el algoritmo fuera de línea; (b) en algoritmo en línea.

En el caso del algoritmo adaptativo (Sec. 5.3.1) no se produce este error ya que el vector representante de la clase se obtiene como media de las observaciones incidentes en dicha clase que se aproximará más al eje de la distribución (Figura 5.12b).

5.6 DISCUSIÓN DE RESULTADOS. CARACTERIZACIÓN DE LA DISTRIBUCIÓN DE FRECUENCIAS ANGULARES.

Una vez descrito el principio del método (Sec. 5.2), analizados los algoritmos propuestos (Sec. 5.3) y obtenidos los resultados experimentales (Sec. 5.4) se pueden abstraer ciertas propiedades acerca de la forma de la distribución de puntos del espacio de mezcla, los parámetros de los que depende, y como estos afectan a la convergencia del procedimiento.

Según se probó en la Sec.5.2, los máximos de la distribución de frecuencias angulares, α_{M1} y α_{M2} , corresponden a las pendientes de los ejes de la distribución de puntos en el espacio de observaciones, verificándose (5.20):

$$\alpha_{M1} = \text{arctg} \frac{a_{21}}{a_{11}} \quad \alpha_{M2} = \text{arctg} \frac{a_{22}}{a_{12}} \quad (5.34)$$

Donde α_{M1} y α_{M2} , en radianes, pueden relacionarse aproximadamente con las clases ganadoras, c_{M1} y c_{M2} , de las n_c en las que se han dividido los 2 cuadrantes superiores (π radianes) según:

$$\alpha_{M1} \approx c_{M1} \frac{\pi}{n_c} - \frac{\pi}{2n_c} \quad \alpha_{M2} \approx c_{M2} \frac{\pi}{n_c} - \frac{\pi}{2n_c} \quad (5.35)$$

siendo $\frac{\pi}{n_c}$ el ángulo en radianes correspondiente a cada una de las n_c clases (5.19).

Las clases ganadoras, tal como se dijo en la Sec. 5.4 (5.21) pueden obtenerse en función de los ángulos α_{M1} y α_{M2} :

$$c_{M1} \approx \text{ent} \left[\frac{n_c}{\pi} \cdot \alpha_{M1} \right] + 1 \quad c_{M2} \approx \text{ent} \left[\frac{n_c}{\pi} \cdot \alpha_{M2} \right] + 1 \quad (5.36)$$

Con las relaciones (5.34), (5.35) y (5.36) quedan caracterizadas las clases ganadoras en función de los elementos de la matriz de mezcla.

La distribución de frecuencias angulares también viene determinada por los valores de los “picos” de la distribución (Figuras 5.10 a 5.12, y 5.14 a 5.19), es decir, por el número de observaciones, $f(c_{M1})$ y $f(c_{M2})$, que se producen dentro de las clases c_{M1} y c_{M2} . Obviamente, el procedimiento de separación de señales propuesto en este capítulo será más preciso cuanto más iguales sean ambos picos.

Es interesante, por tanto, relacionar las frecuencias en las clases ganadoras, $f(c_{M1})$ y $f(c_{M2})$, con los elementos de la matriz de mezcla. Para facilitar este análisis, se van a considerar distribuciones gamma bilaterales, con $\sigma=0.1$, en distintas situaciones en las que las clases ganadoras, c_{M1} y c_{M2} , van a ser fijas. Decir que las clases son fijas equivale a decir que las pendientes a_{21}/a_{11} y a_{22}/a_{12} son constantes, o lo que es lo mismo, que hay que utilizar distintas matrices de mezcla pero todas ellas *similares* a una matriz $\mathbf{A}_0=(a_{ij})$, en el sentido de la Definición 2.1 (2.10). En definitiva, las matrices a considerar son de la forma:

$$\mathbf{A} = \begin{pmatrix} k_1 a_{11} & k_2 a_{12} \\ k_1 a_{21} & k_2 a_{22} \end{pmatrix} = \mathbf{A}_0 \begin{pmatrix} k_1 & 0 \\ 0 & k_2 \end{pmatrix} \quad (5.37)$$

donde los valores de a_{ij} van a ser fijos, variando únicamente los valores de k_1 y k_2 . De esta forma, según (5.34) y (5.37) las pendientes y las clases ganadoras no variarán ya que:

$$\frac{k_1 a_{21}}{k_1 a_{11}} = \frac{a_{21}}{a_{11}} = \text{tg}(\alpha_{M1}) = \text{cte.} \quad \frac{k_2 a_{22}}{k_2 a_{12}} = \frac{a_{22}}{a_{12}} = \text{tg}(\alpha_{M2}) = \text{cte.} \quad (5.38)$$

Por otra parte, para poder analizar adecuadamente la redistribución de puntos entre las dos clases máximas, interesa que estas estén lo más separadas posible, es decir, que se verifique:

$$c_{M2} - c_{M1} \approx \frac{n_c}{2} \quad (5.39)$$

Una matriz canónica que verifica la condición (5.39) es:

$$\mathbf{A}_d = \begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix} \approx \mathbf{A}_8 \quad (5.40)$$

En efecto, con $n_c=180$, se tiene (5.34):

$$\alpha_{M1} = \arctg(1) = 0.79; \quad \alpha_{M2} = \arctg(-1) = -0.79 \quad (5.41)$$

y por tanto, según (5.36) se verifica que $c_{M1} = 46$ y $c_{M2} = 136$, cumpliéndose la condición (5.39) ($136 - 46 = n_c/2$).

En consecuencia, la matriz (5.37) con A_0 dada por (5.40) constituye, un conjunto de matrices de mezcla adecuado para estudiar la redistribución de puntos entre las clases máximas:

$$A = \begin{pmatrix} k_1 & -k_2 \\ k_1 & k_2 \end{pmatrix} \approx A_8 \quad (5.42)$$

En la Figura 5.23 se representa el espacio de las fuentes y su distribución de frecuencias angulares para dos señales con función de distribución de probabilidad gamma bilateral con $\sigma=0.1$ y $M=100.000$ muestras, y con matriz de mezcla (5.40).

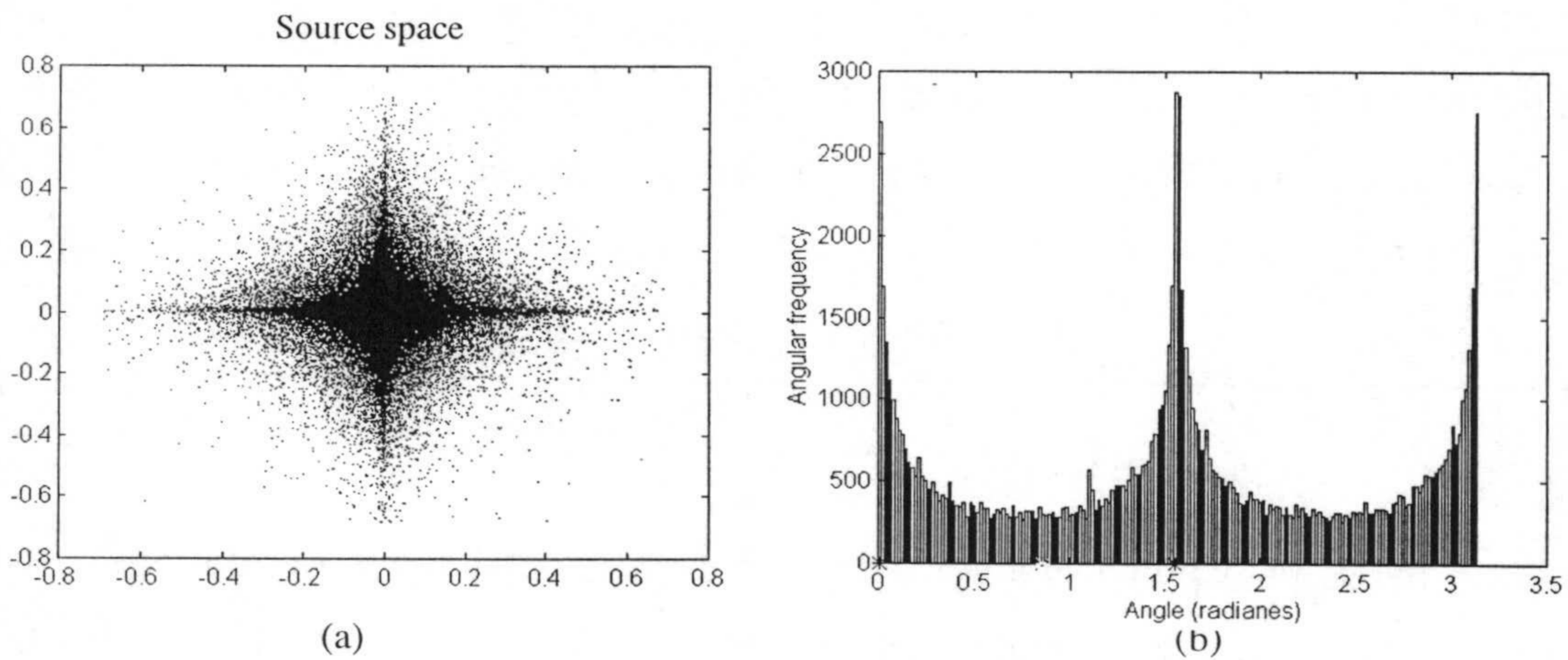


Figura 5. 23 (a) *Espacio fuente* y (b) *Distribución de frecuencias angulares de dos señales aleatorias con función de probabilidad gamma bilateral, $\sigma=0.1$, $NM=100.000$, $n_c=180$, y mezcla A_8 .*

En la Figura 5.24 se incluyen las distribuciones espaciales y las distribuciones de frecuencias angulares obtenidas para distintas matrices del tipo A_8 (5.42).

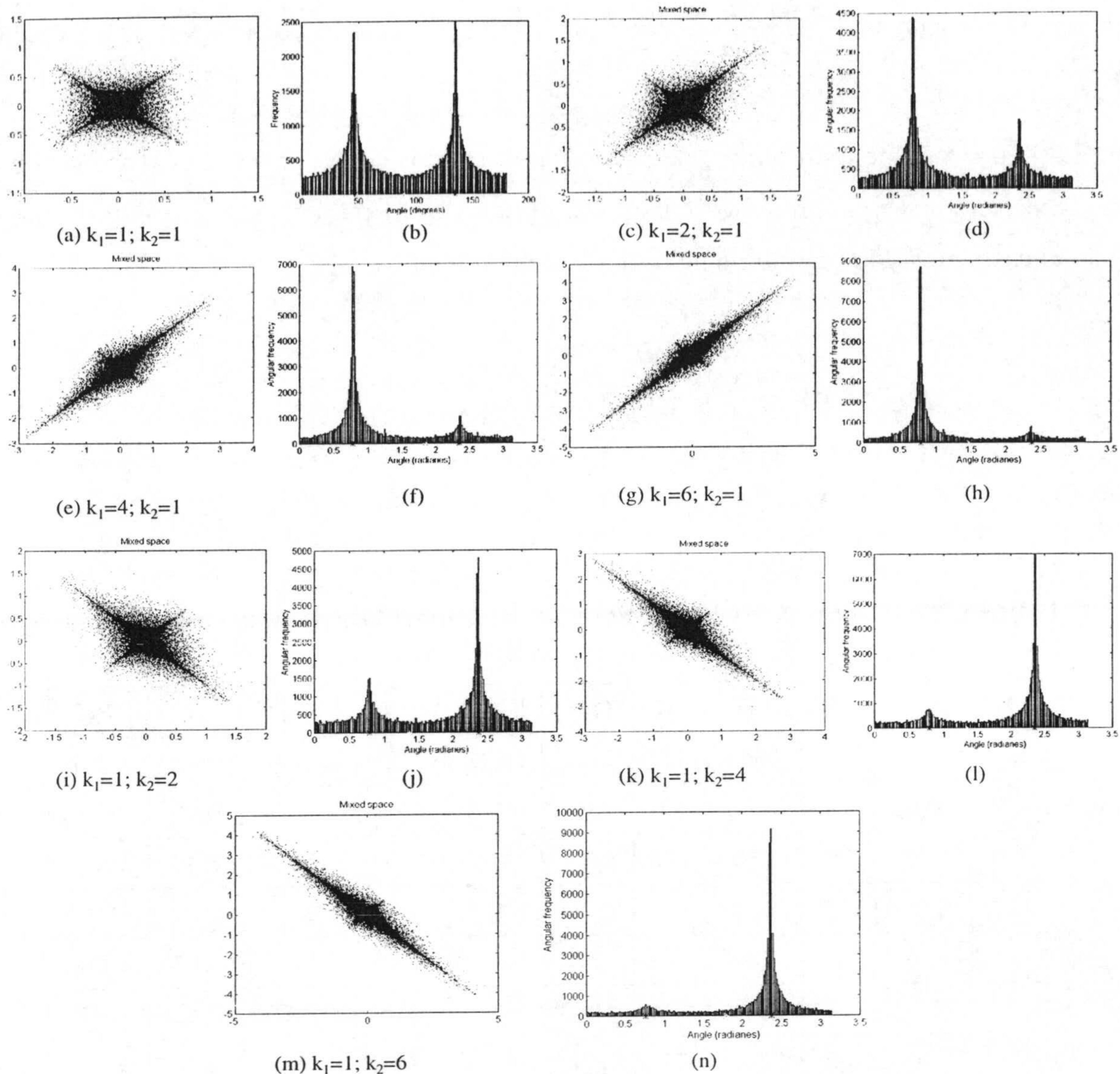


Figura 5. 24. *Espacio de observaciones y distribución de frecuencias angulares para distintos valores de k_1 y k_2 ($A_8=[k_1 -k_2; k_1 k_2]$), fuentes: gamma bilateral con $\sigma=0.1$, $M=100.000$, $n_c=180$.*

Claramente se observa la influencia de la relación k_1/k_2 en la redistribución del número de puntos entre las clases ganadoras. Para abstraer mejor los resultados se listan en la Tabla 5.10 los valores de k_1/k_2 y los correspondientes al valor $t_{M1}=f_{M1}/(f_{M1}+f_{M2})$ que indica la proporción de puntos que caen en las proximidades del primer máximo con respecto al número total de puntos en las proximidades de los dos máximos. En dicha tabla f_{M1} y f_{M2} representan el número de puntos en las proximidades del primer y segundo máximo respectivamente:

$$f_{M1} = \frac{f(44) + f(45) + f(46)}{3} \quad f_{M2} = \frac{f(135) + f(136)}{2} \quad (5.43)$$

En la Figura 5.25 se representan gráficamente los puntos $(t_{M1}, k_1/k_2)$ y la curva resultante de hacer un ajuste por mínimos cuadrados utilizando un polinomio de segundo grado. Para el caso considerado se obtiene la siguiente relación:

$$t_{M1} = -0.39 \cdot \left(\frac{k_1}{k_2}\right)^2 + 0.37 \cdot \left(\frac{k_1}{k_2}\right) + 0.041 \quad (5.44)$$

$$t_{M2} = 1 - t_{M1}$$

Tabla 5. 10 Valores de los parámetros t_{M1} y t_{M2} para los casos considerados en la Figura 5.10.

K1	K2	K1/K2	f(44)	f(45)	f(46)	f(135)	f(136)	f _{M1}	f _{M2}	t _{M1}	t _{M2}
1	1	1	1740	2641	2690	2779	2905	2357	2842	0,45	0,55
1	2	0,5	1193	1469	1510	4357	4790	1391	4574	0,23	0,77
1	4	0,25	747	736	776	6974	---	753	6974	0,10	0,90
1	6	0,17	573	450	---	8464	9151	512	8808	0,05	0,95
2	1	2	2421	4398	---	1772	---	3410	1772	0,66	0,34
4	1	4	3364	6912	---	1069	---	5138	1069	0,83	0,17
6	1	6	4026	8586	8718	777	805	7110	791	0,90	0,10

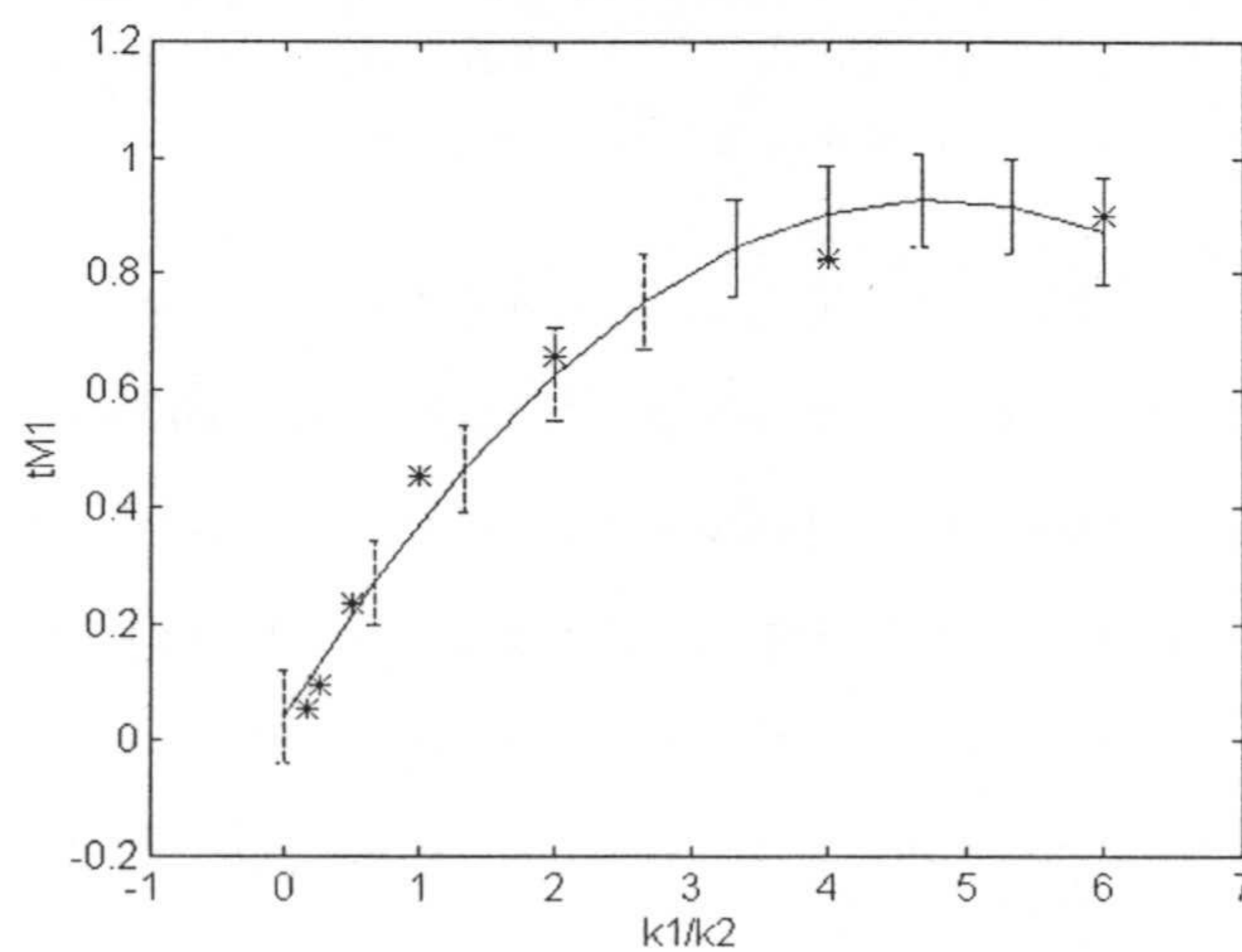


Figura 5. 25. Representación de los valores $(t_{M1}, k_1/k_2)$ de la Tabla 5.10 y ajuste a un polinomio de segundo grado.

Como conclusiones de lo indicado hasta ahora en esta sección, se puede establecer que:

1. Los ejes de la distribución forman con el eje e_1^+ los ángulos α_{M1} y α_{M2} dados por (5.34);
2. La distribución original de los puntos de las fuentes (Figura 5.23a) se deforma de manera que (Figura 5.24) el número de puntos alrededor de cada uno de los máximos se redistribuye dependiendo de las relaciones entre los valores $(k_1 a_{11}, k_1 a_{21})$ con respecto a $(k_2 a_{12}, k_2 a_{22})$; en efecto, conforme (k_1/k_2) aumenta o disminuye, se reduce el número de puntos alrededor de uno u otro máximo de la distribución, respectivamente, dificultándose la aplicación el método aquí propuesto. La situación óptima se presenta cuando $k_1=k_2$.

La primera conclusión se justificó analíticamente en las Sec.2.2 y 5.2. La segunda conclusión puede concretarse más y justificarse analíticamente con las consideraciones que se hacen a continuación.

Supóngase, para simplificar, que:

$$\begin{aligned} 0 \leq s_1(t) \leq 10 \\ 0 \leq s_2(t) \leq 10 \\ a_{ij} > 0 \end{aligned} \tag{5.45}$$

Entonces, los vértices del espacio de observaciones (Figura 5.26) tendrán las siguientes coordenadas (Sec. 2.2):

$$\begin{aligned} v_{mm} &= (0,0) \\ v_{mM} &= (10 k_2 a_{12}, 10 k_2 a_{22}) \\ v_{Mm} &= (10 k_1 a_{11}, 10 k_1 a_{21}) \\ v_{MM} &= (10 k_1 a_{11} + 10 k_2 a_{12}, 10 k_1 a_{21} + 10 k_2 a_{22}) \end{aligned} \tag{5.46}$$

y, en consecuencia (Figura 5.26):

- La posición del vértice v_{mM} viene determinada por:
 - el valor $k_2 a_{12}$ en el sentido e_1

- el valor $k_2 a_{22}$ en el sentido e_2
- La posición del vértice v_{Mm} viene determinada por:
 - el valor $k_1 a_{11}$ en el sentido e_1
 - el valor $k_1 a_{21}$ en el sentido e_2

Por lo tanto, si los valores a_{ij} son constantes para unas fuentes determinadas (s_{jM} y s_{jm} determinadas), los valores de k_1 y k_2 determinarán la distancia entre aristas paralelas (Figura 5.26):

- k_1 determinará la distancia, d_{Mm} , entre $0v_{mM}$ y $v_{Mm}v_{MM}$
- k_2 determinará la distancia, d_{mM} , entre $0v_{Mm}$ y $v_{mM}v_{MM}$

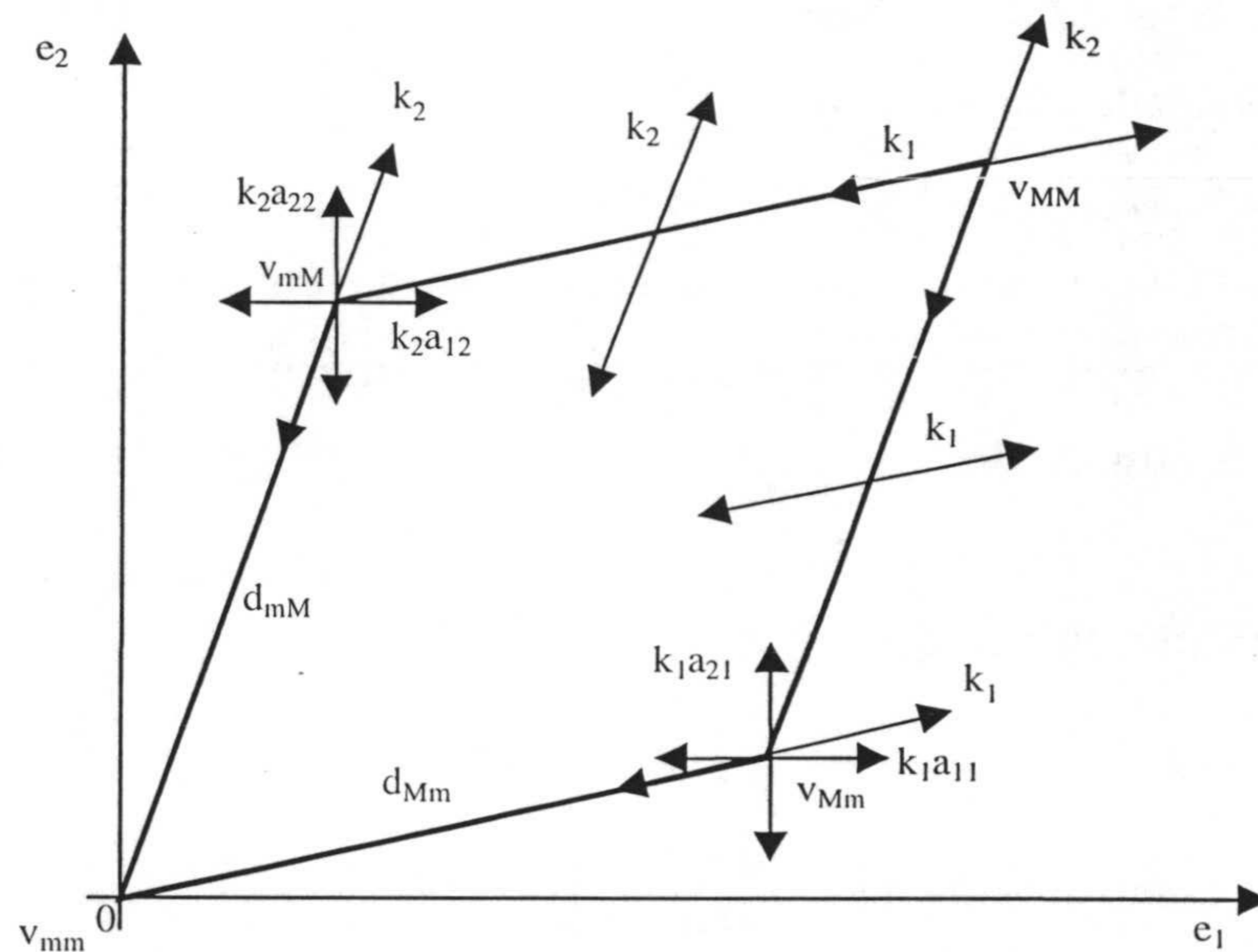


Figura 5. 26 Influencia de los valores de los elementos de A en la posición de los vértices del paralelepípedo que contiene el espacio de observaciones.

En la Figura 5.27 se observa como se transforma el paralelepípedo que contiene el espacio de observaciones para distintos valores de k_1/k_2 .

La pérdida de simetría del paralelepípedo provoca una deformación en la distribución de puntos (compárense la Figura 5.27 con la 5.24, aunque correspondan a distintas matrices de mezcla, A_1 y A_8 respectivamente).

En consecuencia, la relación k_1/k_2 determina la relación entre los tamaños de las aristas, d_{Mm}/d_{mM} , y por tanto la longitud relativa de los ejes de la distribución de puntos en el

espacio de observaciones (ver Figura 5.24) con respecto a la longitud de los ejes de la distribución original de los puntos de las fuentes (Figura 5.23a). Por otra parte, la relación entre la longitud de los dos ejes de la distribución de puntos de mezcla influye en el número de puntos abarcados por los ángulos de las clases de frecuencias máxima (Figura 5.24).

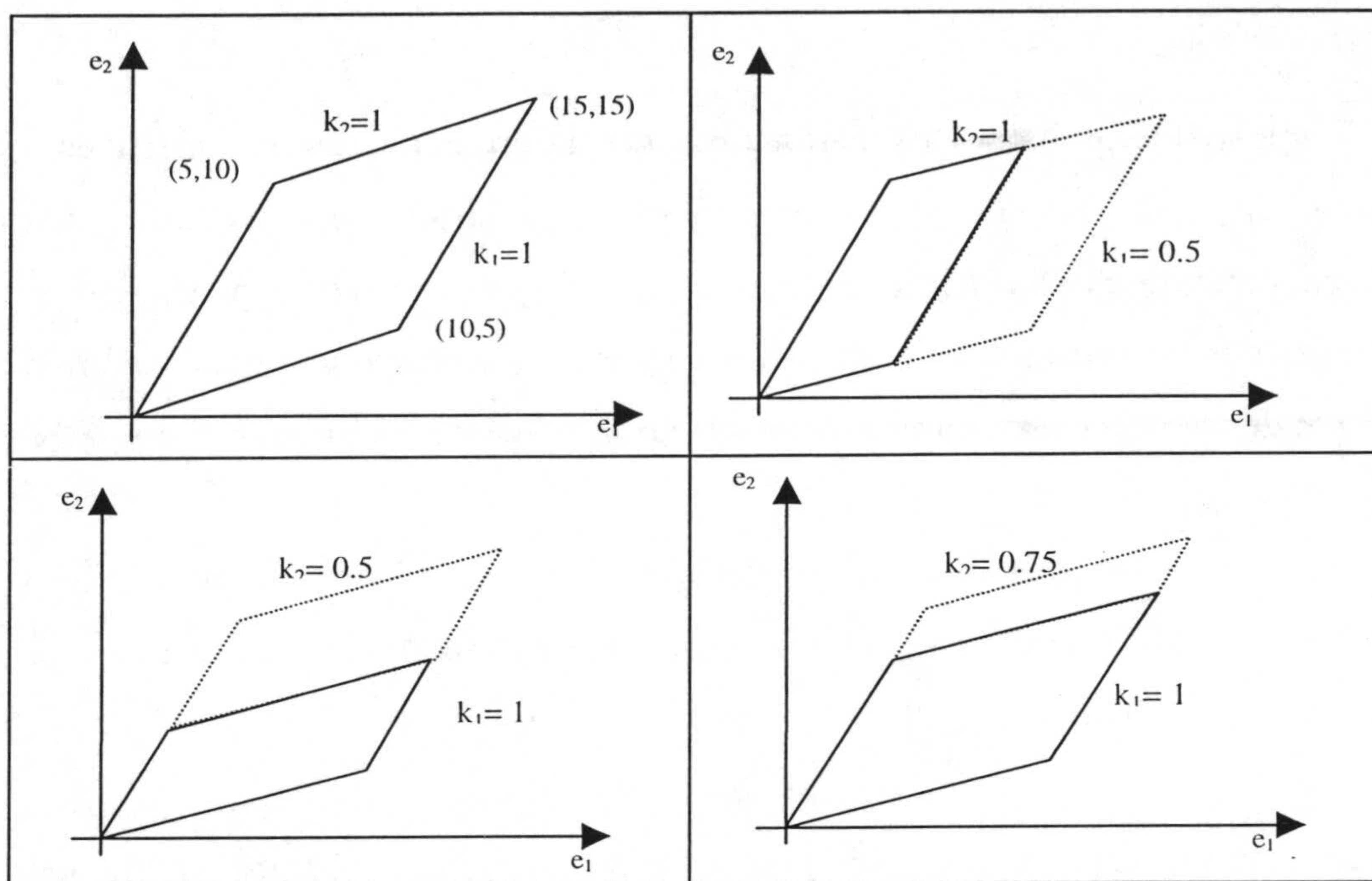


Figura 5. 27. *Modificaciones del paralelepípedo que contiene el espacio de observaciones para distintos valores de k_1 y k_2 (matriz A , $s_{1M}=s_{2M}=10$, $s_{1m}=s_{2m}=0$).*

Con el procedimiento propuesto en este capítulo, la separación será tanto más eficiente cuanto menos deformada sea la distribución de puntos del espacio de la fuentes (Figura 5.23a) y por tanto, cuanto más similares sean las longitudes d_{Mm} y d_{mM} (que a partir de ahora, para simplificar la notación se denominarán d_1 y d_2 respectivamente). Definimos un *parámetro de deformación*, D , que es 1 cuando la diferencia entre los ejes es máxima (d_1 o $d_2=0$) y 0 cuando son iguales ($d_1 = d_2$):

$$D = \frac{|d_1 - d_2|}{d_1 + d_2}; \quad 0 \leq D \leq 1 \quad (5.47)$$

siendo:

$$\begin{aligned} d_1 &= k_1 \sqrt{a_{11}^2 + a_{21}^2} \\ d_2 &= k_2 \sqrt{a_{12}^2 + a_{22}^2} \end{aligned} \quad (5.48)$$

Si se verifica que $a_{11}^2 + a_{21}^2 = a_{12}^2 + a_{22}^2$ como sucede con las matrices A_1 , A_2 y A_3 , se tiene que:

$$D = \left| \frac{k_1 - k_2}{k_1 + k_2} \right|; \quad 0 \leq D \leq 1 \quad (5.49)$$

En la Tabla 5.11 se listan los parámetros de deformación de las matrices A_1 a A_8 utilizadas en este capítulo. Obsérvese que los máximos en las distribuciones de frecuencias angulares para las matrices A_1 , A_2 y A_3 en las Figuras 5.10, 5.11 y 5.12 no son iguales por no serlo los valores de s_{jM} y s_{jm} de las correspondientes fuentes (para que fuesen iguales debería verificarse $s_{1M} = s_{2M}$, $s_{1m} = s_{2m}$).

Cuanto menor sea el valor de D , menos deformado estará el paralelepípedo del espacio de mezclas y más fácil será la separación (con menor número de muestras se obtendrán resultados satisfactorios).

Otro factor que influye en la velocidad de convergencia del procedimiento de estimación de la matriz \mathbf{A} es la separación entre las dos clases c_{M1} y c_{M2} donde se obtienen las frecuencias angulares máximas. Obviamente cuanto más separadas estén las clases menos se solapará la influencia de un máximo sobre el otro. La separación máxima (óptima) será $n_c/2$. Es posible definir un *parámetro de cercanía*, C , que sea 1 cuando las clases máximas coincidan ($c_{M1} = c_{M2}$) y 0 cuando la separación sea máxima ($|c_{M1} - c_{M2}| = n_c/2$) de la siguiente forma:

$$C = 1 - \frac{2}{n_c} |c_{M1} - c_{M2}| = 1 - \frac{2}{\pi} \left[\arctg\left(\frac{a_{22}}{a_{12}}\right) - \arctg\left(\frac{a_{21}}{a_{11}}\right) \right] \quad 0 \leq C \leq 1 \quad (5.50)$$

En la Tabla 5.10 se incluyen los valores del parámetro de cercanía para las 8 matrices A_1 a A_8 .

Tabla 5. 11 *Parámetros de deformación y proximidad correspondientes a las matrices A₁ a A₈.*

	a ₁₁	a ₁₂	a ₂₁	a ₂₂	c _{M1}	c _{M2}	c _{M1} -c _{M2}	d ₁	d ₂	D	C
A ₁	1	0,5	0,5	1	27	64	37	1,5	1,5	0	0,59
A ₂	0,8	0,1	0,2	0,9	14	84	70	1	1	0	0,22
A ₃	0,8	0,4	-0,5	0,9	67	148	81	1,3	1,3	0	0,1
A ₄	0,3	-0,1	-0,4	-1	127	85	42	1,1	0,7	0,22	0,53
A ₅	-0,7	0,1	0,1	-1	172	96	76	1,1	0,8	0,16	0,16
A ₆	-1	-0,1	-0,8	0,1	39	96	57	0,2	1,8	0,8	0,37
A ₇	0,6	0,7	-0,3	1,4	154	64	90	2,1	0,9	0,4	0
A ₈	1	-0,98	1	1	45	135	90	1,98	1,97	0,003	0

La cercanía resulta ser mayor para las matrices A₁ y A₄, que tienen C=0.59 y 0.53 respectivamente, y la menor es para A₇ y A₈ con un valor de C=0. Estos resultados concuerdan con las representaciones de las Figuras 5.10 y 5.11.

Cuando la cercanía es grande, los puntos correspondientes a las clases próximas a la de uno de los máximos, se superponen a las clases próximas a la del otro (y viceversa) produciéndose un desplazamiento de los máximos, aproximándose ambos. Así, en los resultados de las Tablas 5.5 y 5.6 (comparar con la Tabla 5.3) se produce el desplazamiento de los máximos (15 en vez de 14, 173 en lugar de 172 y 96 en vez de 97, respectivamente).

Según se indicó en la Sec.5.5, el efecto del desplazamiento del máximo en una clase provoca un error de π/n_c radianes (0.018 radianes si $n_c=180$) en la determinación del ángulo α_M , y el consiguiente error en la estimación de los elementos de **A**.

El desplazamiento de los máximos se puede comprender mejor con ayuda de la Figura 5. 28, donde se muestra el efecto de superponer (sumar) dos funciones de distribución de probabilidad gaussiana con desviaciones estándar $\sigma_1=16$, $\sigma_2=14$ y medias $\mu_1=5$, $\mu_2=40$. Se observa claramente que los dos máximos (“jorobas”) de las funciones aisladas se desplazan al superponerlas aproximándose uno al otro (máximo de la gaussiana 1 = 0.0285, máximo de la gaussiana 2 = 0.0249, máximos de la suma de gaussianas = 0.0266 y 0.0312).

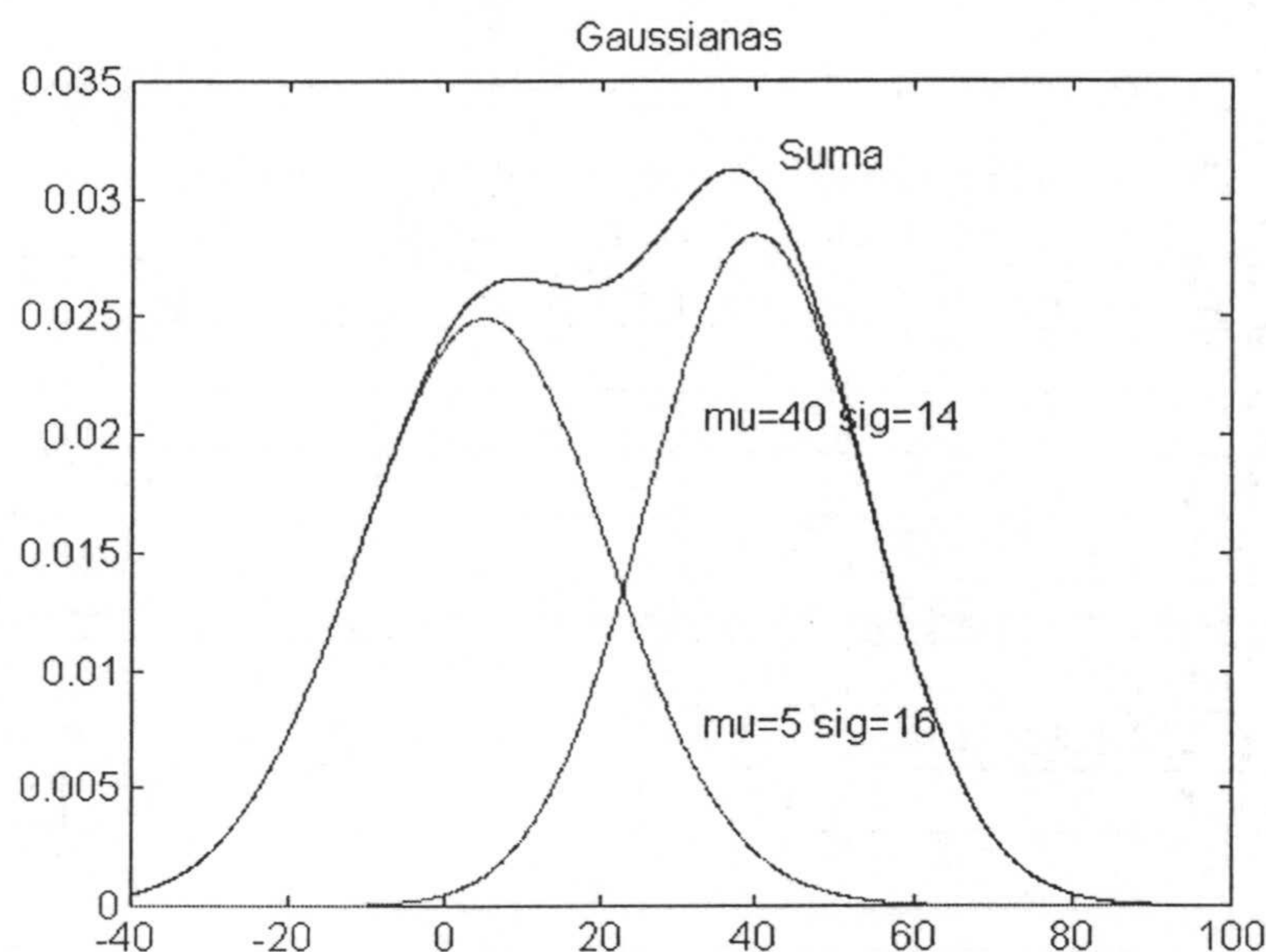


Figura 5. 28 Efecto de la superposición (suma) de dos funciones de distribución de probabilidad gaussianas ($\sigma_1=16$, $\mu_1=5$ y $\sigma_2=14$, $\mu_2=40$).

Como conclusión de esta sección se puede afirmar que un inconveniente del método propuesto en este capítulo es que no es *equivariante* en el sentido de que sus prestaciones dependen de los valores concretos de los elementos de la matriz de mezcla (Sec.1.14.4) [CAR96], [AMA95a]. A cada matriz **A** se le puede asociar dos parámetros: la deformación y la proximidad; cuanto más pequeños sean los valores de estos parámetros más rápida y precisamente se obtendrá la separación de la mezcla.

5.7 EFECTOS DEL RUIDO DE LAS MEZCLAS EN LA CONVERGENCIA DEL PROCEDIMIENTO PROPUESTO.

En esta sección se analiza la influencia del ruido sobre las señales mezcladas y cómo aquel afecta a las prestaciones del procedimiento propuesto.

Tal como se indicó en la Sec.2.1.2 los efectos del ruido sobre una señal, en general, se pueden estudiar adecuadamente considerando un modelo aditivo con *ruido blanco* (señal aleatoria donde las muestras no están correlacionadas unas con otras y por tanto, con valor medio nulo [KUN96], [BUR98]) de forma que:

$$e(t) = A \cdot s(t) + n(t) \tag{5.51}$$

donde $\mathbf{n}(t)=(n_j(t))^T$, siendo $n_j(t)$ el ruido blanco que actúa sobre la mezcla $e_j(t)$. El ruido blanco que suele considerarse, consiste en una señal aleatoria con distribución de probabilidad uniforme o gaussiana (Sec.A1.3.1 y A2.3.2).

Desde un punto de vista físico, la superposición del ruido en las mezclas suele producirse en los propios sensores y puede ser, por ejemplo, de origen térmico.

El entorno de simulación que se ha desarrollado (Capítulo 3) permite añadir a las mezclas ruido tanto de tipo gaussiano como de tipo uniforme. En el primer caso el programa solicita al usuario la desviación estándar (σ) y en el segundo caso el porcentaje o nivel de ruido (r) que se desea para la señal final. Es fácil transformar el porcentaje de ruido, r , en relación señal ruido de la mezcla, en decibelios (SNR):

$$SNR = 20 \cdot \log\left(\frac{100-r}{r}\right) = 20 \cdot \log\left(\frac{100}{r} - 1\right) \text{ dBs} \quad (5.52)$$

A continuación se analizan los efectos del ruido, modelado según (5.51), en la eficiencia del procedimiento de separación propuesto en este capítulo.

En la Figura 5.29a se muestra el espacio de mezclas y su distribución en frecuencias angulares (Figura 5.29b) de una mezcla A_4 de dos señales gamma bilateral, sin superponer ruido. En las Figuras 5.29c y d se incluyen las mismas representaciones, respectivamente, pero cuando se ha añadido ruido uniforme con $r=3\%$ ($SNR \approx 30$ dBs).

El ruido desplaza aleatoriamente las observaciones hacia arriba y hacia abajo (componente $n_2(t)$) y hacia la izquierda o derecha (efecto de la componente $n_1(t)$), “difuminando” la forma del espacio de mezclas (Figura 5.29a y c) y suavizando o aplanando los picos de la distribución de frecuencias angulares (obsérvese en 5.29b y d como varían los valores máximos de ≈ 16.000 y 8.000 a ≈ 6.000 y 2.000 , respectivamente). Pero como el ruido es de tipo blanco tiene media 0 con lo que por término medio y considerando un número, m , de muestras suficientes, las direcciones de los ejes de la distribución no variarán notablemente.

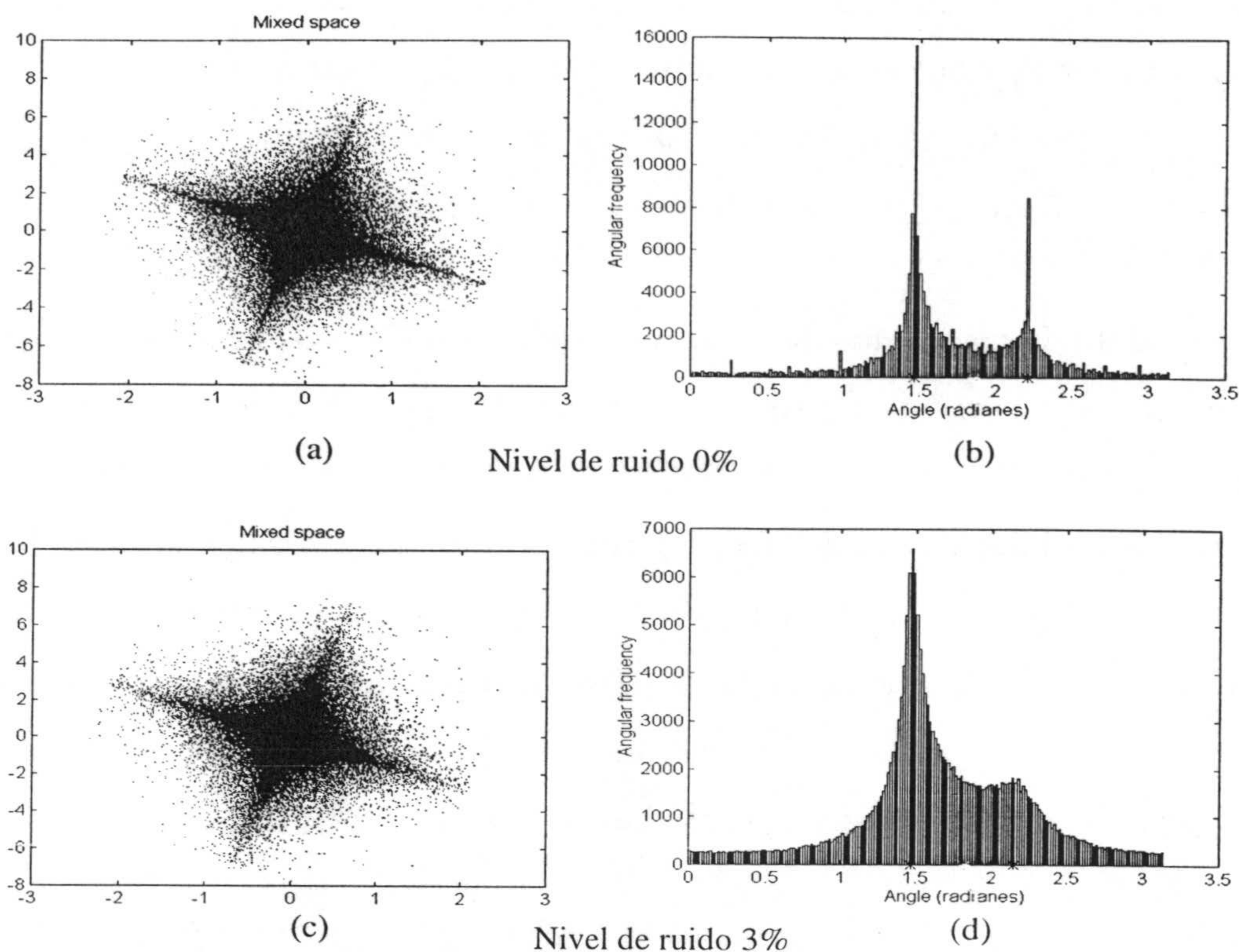


Figura 5.29 *Espacio de mezclas y distribución de frecuencias angulares de una mezcla A_4 de dos señales gamma bilateral sin ruido (a,b) y con un 3% de ruido (c, d).*

El efecto descrito en el párrafo anterior se pone más claramente de manifiesto con ayuda de la Figura 5.30, donde se observa la variación de la distribución de frecuencias angulares para la matriz de mezcla A_3 con ruido gaussiano y desviaciones estándar $\sigma=0, 0.1, 0.2, 0.4, 0.5$ y 0.6 . En estos casos los efectos del ruido sobre las prestaciones del procedimiento pueden considerarse pequeños ya que el error absoluto medio en la estimación de los elementos de A_3 varía de $\varepsilon_{\sigma=0} = 0.0054$ a $\varepsilon_{\sigma=0.6} = 0.0184$. Consideramos que la separación es correcta cuando $\varepsilon_L \leq 0.02$; a partir de este límite de error absoluto desechamos la matriz W que estima la matriz A por imprecisa (Sec.5.5), con lo que con $\sigma = 0.6$ la separación sigue siendo válida ya que se obtiene $\varepsilon=0.0184$.

Para tener una visión analítica y más global de los efectos del ruido (de tipo uniforme), la Tabla 5.12 recoge los errores cuadráticos medios en la estimación de la matriz de mezcla, obtenidos con $m=200.000$ muestras y con las ocho matrices (A_1 a A_8) que se están considerando en este capítulo.

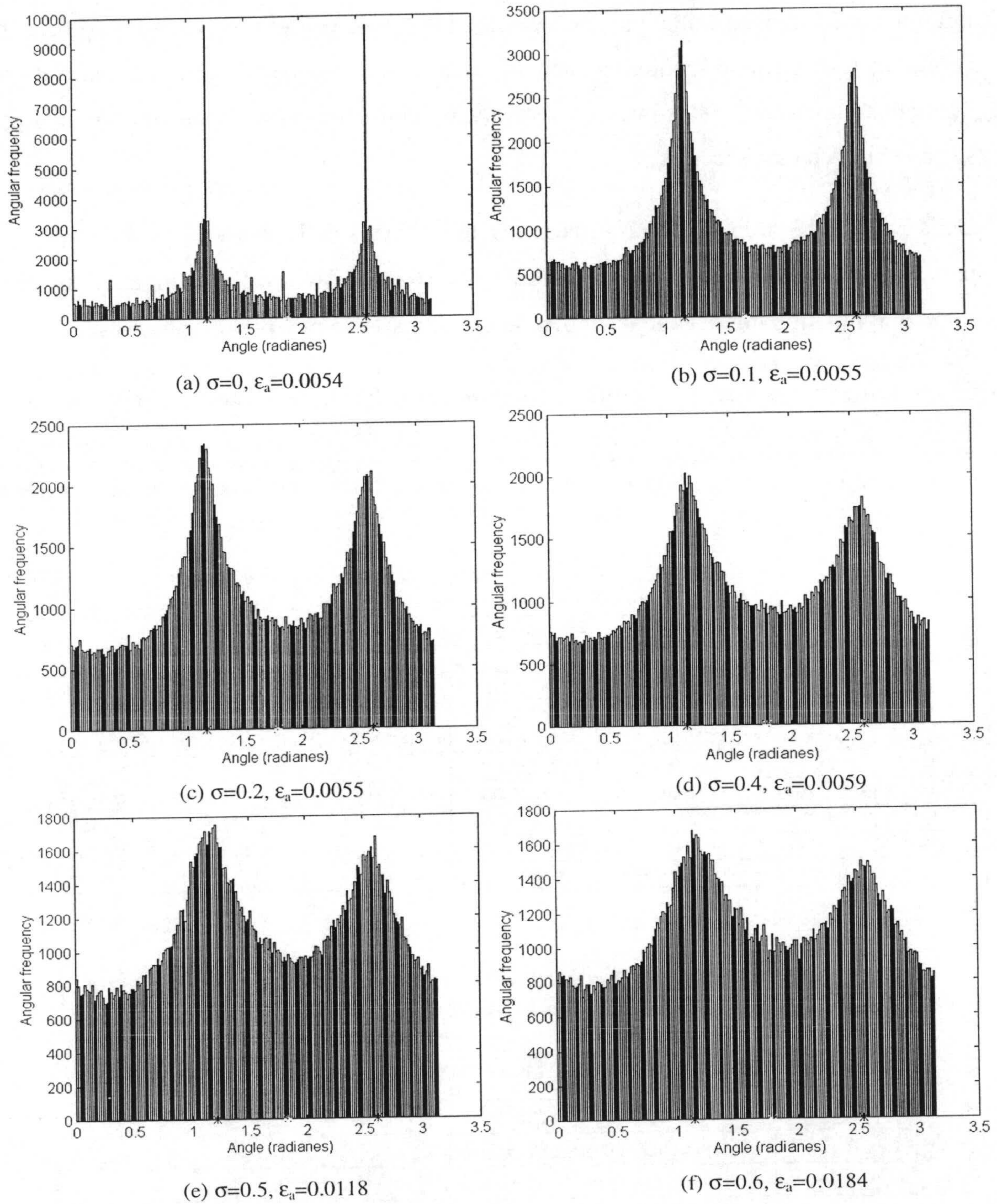


Figura 5. 30. Distribución de frecuencias angulares con ruido gaussiano para distintos valores de σ .

En la Figura 5.30 se representan los resultados de la Tabla 5.12 y se muestra, para cada matriz de mezcla, una recta a la que se ajustan los puntos utilizando el método de los mínimos cuadrados. También se incluyen los intervalos con el 50% de grado o nivel de confianza (por término medio, de cada 100 muestras al menos 50 entrarán dentro de los intervalos indicados). Estos intervalos se incluirán en el resto de ajustes polinómicos que se presentan en esta sección.

En la Tabla 5.13 se incluyen las ecuaciones de las rectas indicadas y el nivel de ruido, r_L , para el que se obtiene un error absoluto de $\varepsilon_L=0.02$. Este último valor indica el nivel de ruido máximo que tolera el algoritmo, caso de utilizar sólo $m=200000$ muestras.

Es conveniente resaltar que la última columna de la Tabla 5.13 indica el nivel de ruido máximo que se admite cuando se consideran $m=200.000$ muestras. Con un número superior de muestras se tolerarían niveles mayores de error, según se pone de manifiesto en los párrafos siguientes.

Tabla 5. 12. Error absoluto obtenido con $m=200.000$, con las ocho matrices (A_1 a A_8) para distintos niveles de ruido (10 medidas en cada caso).

Matriz A	Nivel de ruido uniforme (%)					
	0	1	2	3	4	5
A_1	0.0005	0.0021±0.0032	0.0085±0.0036	0.0183±0.0077	0.0308±0.0141	0.0376±0.0218
A_2	0.0022	0.0029±0.0001	0.0063±0.0028	0.0066±0.0047	0.0128±0.0080	0.0218±0.0098
A_3	0.0054	0.0055±0.0002	0.0055±0.0001	0.0079±0.0041	0.0108±0.0051	0.0160±0.0124
A_4	0.0050	0.0054±0.0001	0.0274±0.0091	0.1105±0.1426	0.5526±0.0372	0.5647±0.0189
A_5	0.0024	0.0030±0.0006	0.0050±0.0023	0.0075±0.0051	0.0137±0.0069	0.0191±0.0098
A_6	0.0017	0.0021	0.0077±0.0022	0.0122±0.0043	0.0175±0.0085	0.0199±0.0109
A_7	0.0006	0.0032±0.0036	0.0102±0.0061	0.0100±0.0057	0.0361±0.0159	0.0309±0.0292
A_8	0.0040	0.0053±0.0009	0.0066±0.0032	0.0088±0.0061	0.0111±0.0067	0.0172±0.0104

Tabla 5. 13 Variación del error absoluto de estimación de A en función del nivel de ruido (uniforme), $m=200.000$.

Matriz A	Ecuación de la recta	Nivel de ruido máximo (r_L)
A_1	$\varepsilon = 0.0032 \cdot r - 0.0012$	6.6 %
A_2	$\varepsilon = 0.0040 \cdot r + 0.0015$	11.6 %
A_3	$\varepsilon = 0.001 \cdot r + 0.005$	30 %
A_4	$\varepsilon = 0.04 \cdot r - 0.04$	1.6 %
A_5	$\varepsilon = 0.001 \cdot r + 0.003$	21.8 %
A_6	$\varepsilon = 0.0015 \cdot r + 0.0013$	12.5 %
A_7	$\varepsilon = 0.0017 \cdot r + 0.0036$	9.7 %
A_8	$\varepsilon = 0.0006 \cdot r + 0.0046$	25.7 %

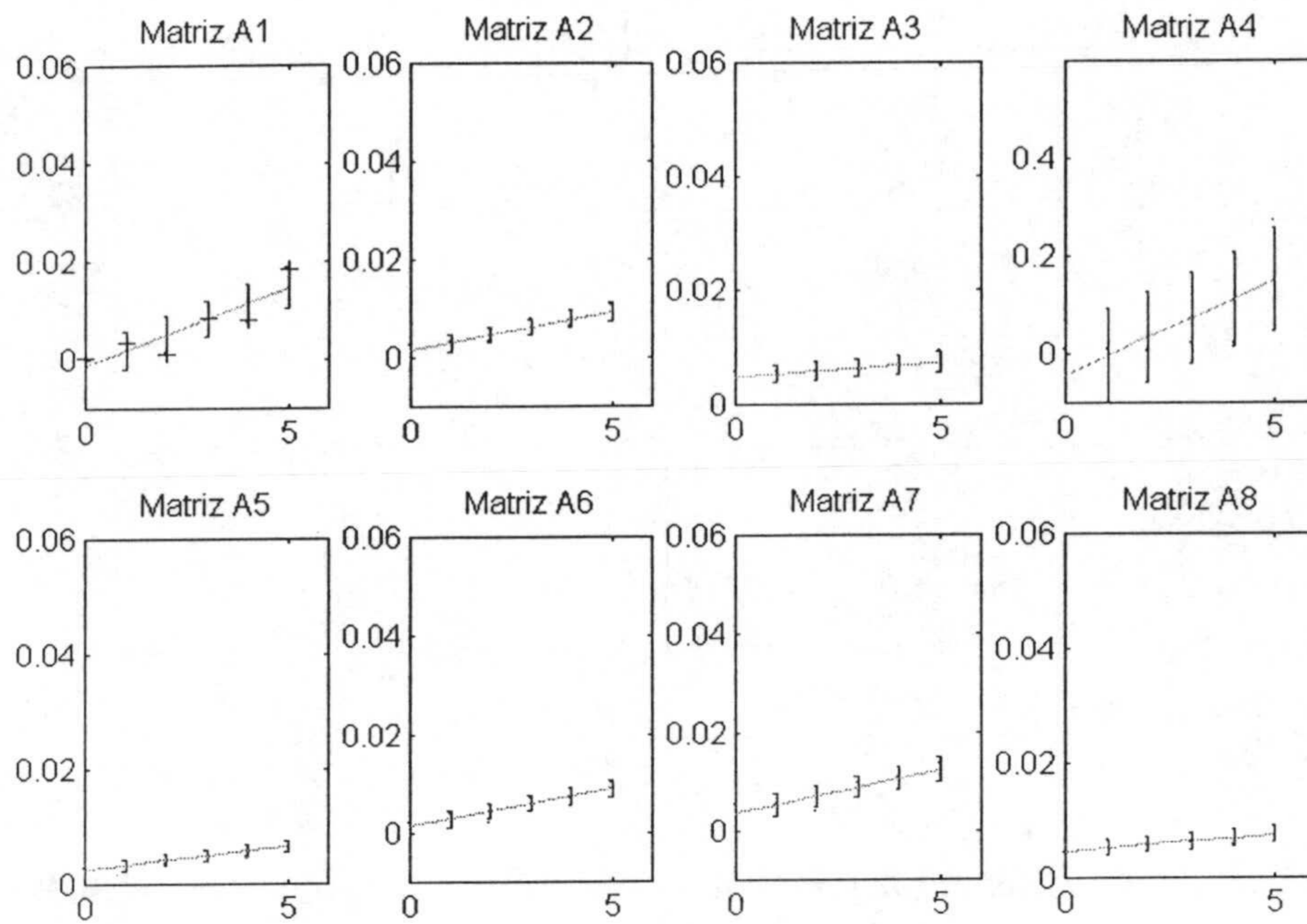


Figura 5. 31 Variación del error absoluto frente al nivel de ruido (0-3.5 %) para distintas matrices de mezcla (A_1 a A_8).

En la Tabla 5.14 y Figura 5.32 se recogen resultados similares para ruido gaussiano (en vez de uniforme).

Tabla 5. 14. Error absoluto obtenido con $m=200.000$, con las ocho matrices (A_1 a A_8) para distintos niveles de ruido gaussiano (una medida en cada caso).

Matriz A	Desviación estándar (σ)						s
	0	0.1	0.2	0.4	0.5	0.6	
A_1	0.0005	0.0055	0.0158	0.0219	0.0334	0.0624	
A_2	0.0024	0.0029	0.0058	0.0199	0.0167	0.0075	0.0102
A_3	0.0046	0.0058	0.0054	0.0054	0.0116	0.0027	0.0180
A_4	0.0046	0.0181	0.0452	0.1691	0.0920	0.1280	0.2426
A_5	0.0021	0.0037	0.0063	0.0113	0.0082	0.0160	0.0368
A_6	0.0016	0.0225	0.0064	0.0252	0.0320	0.0513	0.0426
A_7	0.0005	0.0007	0.0055	0.0109	0.0393	0.2881	0.2692
A_8	0.0032	0.0039	0.0060	0.0130	0.0270	0.0250	0.0285

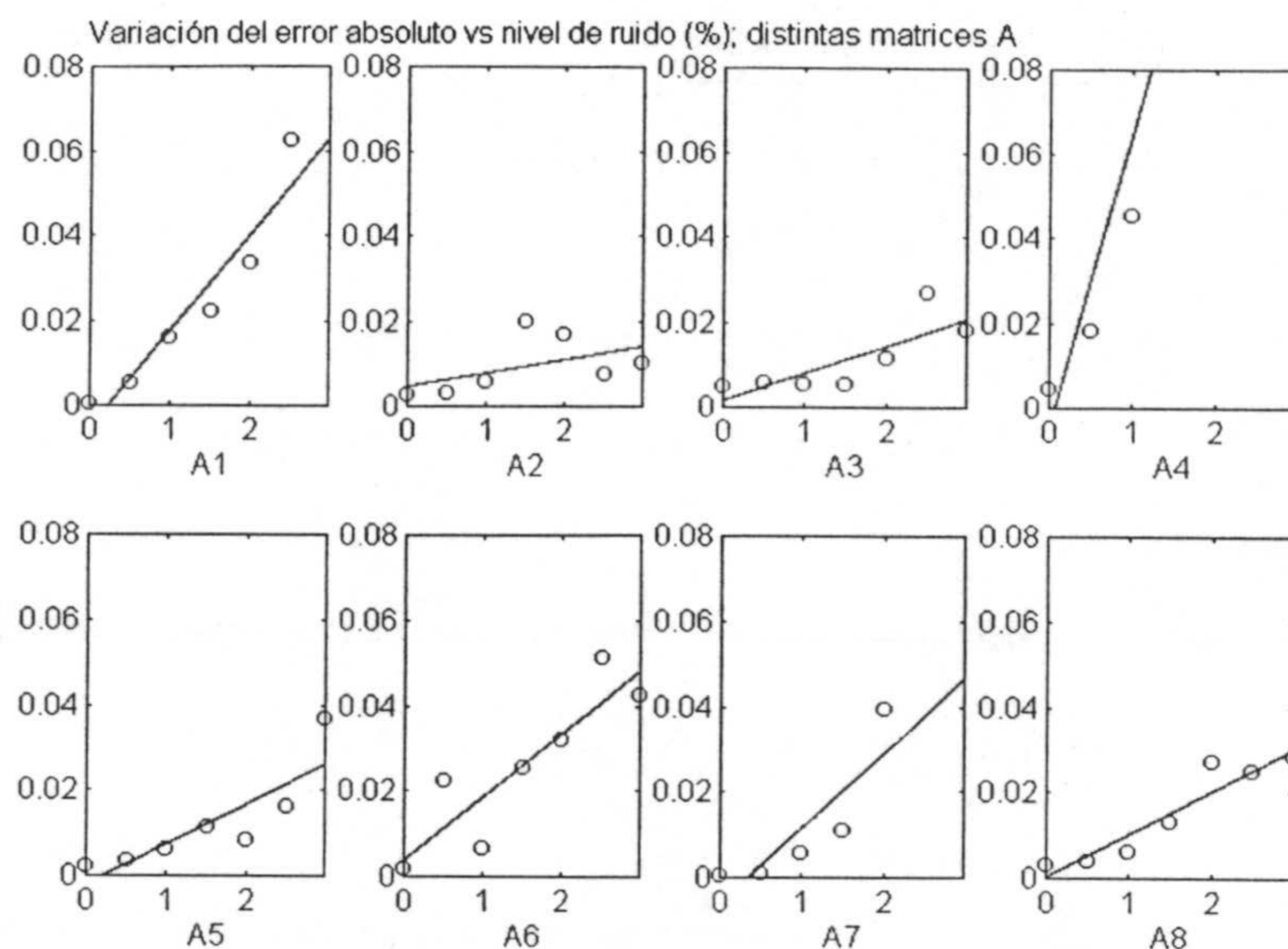


Figura 5.32 Variación del error absoluto con distintas desviaciones estándar de ruido gaussiano (0–0.6 %) para distintas matrices de mezcla ($A_1 - A_8$).

Para completar el análisis de los efectos del ruido se ha estudiado, para distintos niveles del mismo, la influencia del número de muestras (m) en el error medio absoluto de la estimación de la matriz A . Los resultados completos se incluyen en el Apéndice 3, y aquí se presenta como ejemplo matriz A_8 . Los resultados se muestran en la Tabla 5.15.

Tabla 5.15. Evolución del error absoluto con el número de muestras (matriz A_8) para distintos niveles de ruido uniforme (en cada caso se consideran 10 medidas).

m	Nivel de ruido uniforme (%)						
	0	1	2	3	4	5	6
10.000	0.0058	0.0088	0.0127	0.0134	0.0099	0.0162	0.0160
20.000	0.0038	0.0069	0.0079	0.0187	0.0123	0.0121	0.0128
30.000	0.0038	0.0065	0.0085	0.0164	0.0137	0.0077	0.0050
40.000	0.0039	0.0061	0.0068	0.0120	0.0169	0.0170	0.0140
50.000	0.0040	0.0060	0.0063	0.0095	0.0127	0.0109	0.0041
60.000	0.0042	0.0052	0.0075	0.0121	0.0125	0.0129	0.0130
70.000	0.0042	0.0050	0.0060	0.0084	0.0080	0.0057	0.0060
80.000	0.0042	0.0054	0.0072	0.0082	0.0140	0.0134	0.0118
90.000	0.0042	0.0052	0.0061	0.0090	0.0090	0.0086	0.0070
100.000	0.0042	0.0055	0.0054	0.0089	0.0121	0.0165	0.0114
110.000	0.0042	0.0054	0.0061	0.0089	0.0080	0.0075	0.0053
120.000	0.0042	0.0054	0.0053	0.0093	0.0121	0.0178	0.0116
130.000	0.0042	0.0054	0.0049	0.0067	0.0099	0.0102	0.0075
140.000	0.0042	0.0052	0.0053	0.0083	0.0106	0.0146	0.0119
150.000	0.0041	0.0055	0.0049	0.0071	0.0069	0.0097	0.0082
160.000	0.0041	0.0053	0.0054	0.0080	0.0085	0.0149	0.0120
170.000	0.0041	0.0053	0.0055	0.0063	0.0067	0.0084	0.0068
180.000	0.0041	0.0053	0.0066	0.0078	0.0082	0.0151	0.0126
190.000	0.0041	0.0053	0.0055	0.0060	0.0067	0.0088	0.0053
200.000	0.0040	0.0053	0.0066	0.0088	0.0111	0.0172	0.0105

La Figura 5.33 contiene la representación de los puntos de la Tabla 5.15 y las rectas que se ajustan a ellos por mínimos cuadrados. La Tabla 5.16 lista las ecuaciones de las rectas que relacionan el error medio absoluto, ε , con el número de muestras, m , considerado correspondientes a los distintos niveles de ruido, r .

Tabla 5. 16 Rectas que representan la variación del error de estimación, ε , de A_8 con el número de muestras, m , para distintos niveles de ruido, r .

Nivel de ruido r (%)	Dependencia $\varepsilon=f(m)$	m_L
0	$\varepsilon = - 0.17e-8 m + 0.004$	<10.000
1	$\varepsilon = - 0.92e-8 m + 0.007$	<10.000
2	$\varepsilon = - 1.9e-8 m + 0.009$	<10.000
3	$\varepsilon = - 4.3e-8 m + 0.014$	<10.000
4	$\varepsilon = - 2.9e-8 m + 0.013$	<10.000
5	$\varepsilon = 0.32e-8 m + 0.012$	<10.000

En la última columna se incluye el número de muestras, m_L , necesario para tener un error $\varepsilon_L \leq 0.02$, que en todos los casos ha sido menor de 10000 muestras.

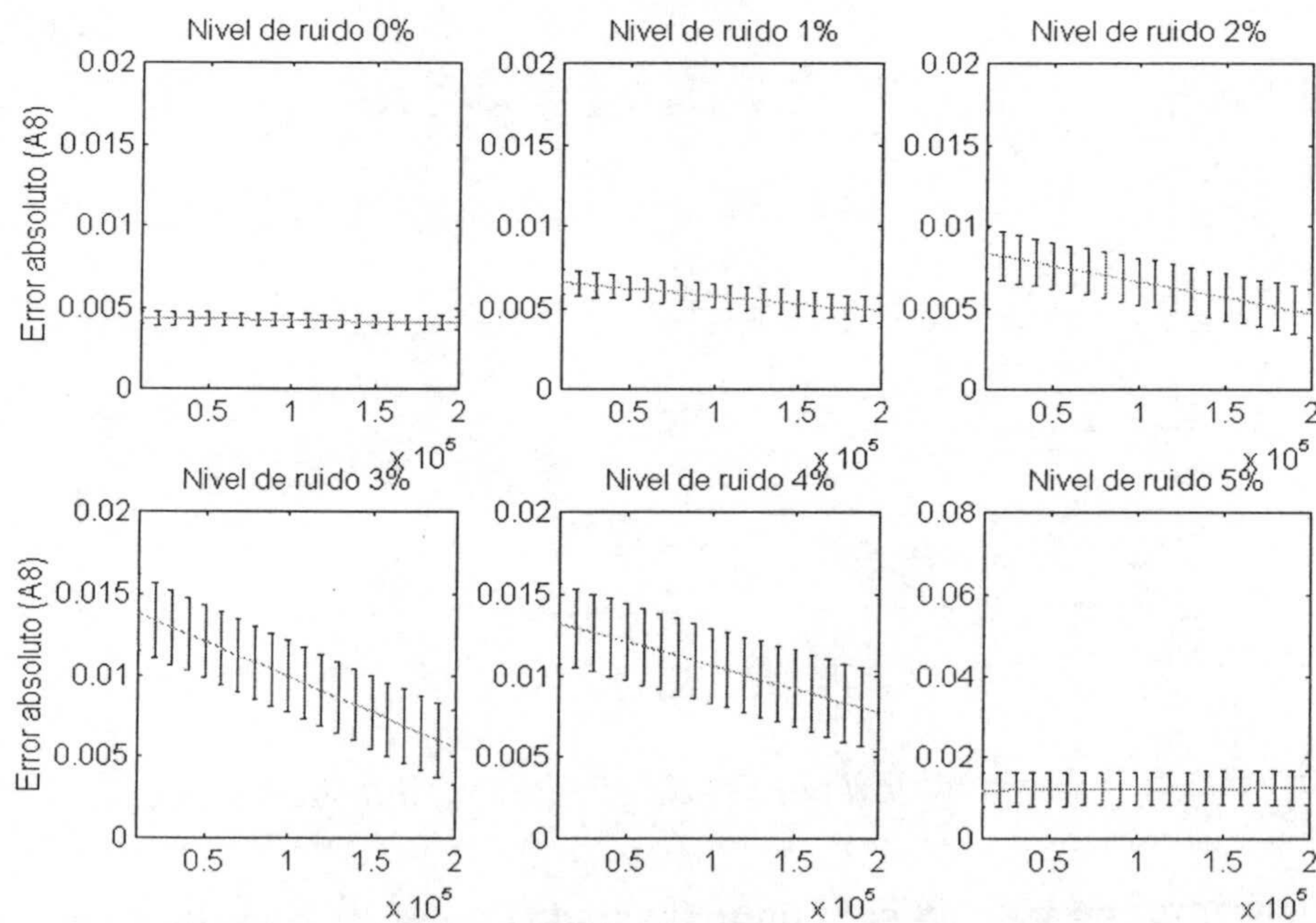


Figura 5. 33 Variación del error absoluto frente al nivel de ruido (0 -3.5 %) para distinto número de muestras con la matriz de mezcla A_8 .

De las ecuaciones de las rectas incluidas en la Tabla 5.16, se puede concluir que, por término medio, para reducir el error en una milésima ($\Delta\varepsilon=0.001$) se necesitan del orden de $\Delta m=100000$ muestras adicionales.

También se ha obtenido para las distintas matrices el número de muestra, m_L , para el que por primera vez se obtiene un error de estimación de la matriz \mathbf{A} menor de $\varepsilon_L \leq 0.02$. Para la matriz \mathbf{A}_8 se tiene la Tabla A3.8.4. En la Figura 5.34, la curva inferior representa los valores medios del número de muestras indicado, $\langle m_L \rangle$, para distintos niveles de ruido, y la curva superior corresponde a los valores medios sumándoles las desviaciones estándar, $\langle m_L(r) \rangle + \sigma_L(r)$. El eje de abscisas, representa el nivel de ruido, r . (En el Apéndice 3 se incluye la misma representación para las 8 matrices que se están considerando en este capítulo).

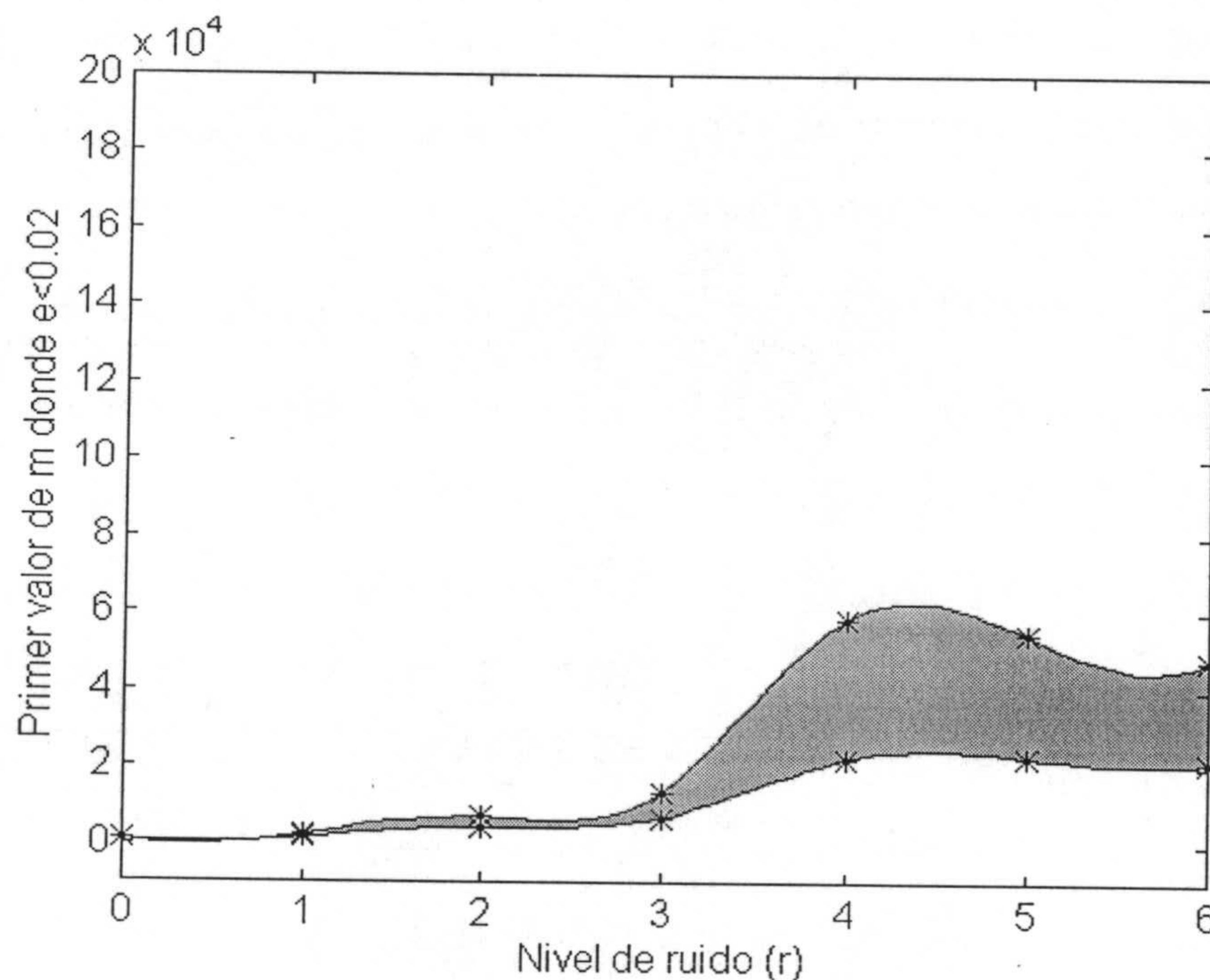


Figura 5. 34. Primer valor de m donde el error absoluto es menor de 0.02.

Ajustando los valores de m_L a un polinomio de grado 2 se obtiene la siguiente ecuación:

$$m_L = 75r^2 + 3896r - 1207 \quad (5.53)$$

De acuerdo con las expresiones (5.53) se puede estimar el número de muestras necesario para realizar adecuadamente ($\varepsilon \leq \varepsilon_L$) la separación de mezclas con un nivel de ruido

predeterminado. Así por ejemplo, el número de muestras necesario para separar una mezcla A_8 , con una relación de ruido $SNR=10$ dBs se puede obtener como se indica a continuación.

De (5.52) se obtiene el nivel de ruido correspondiente:

$$r = \frac{100}{10^{SNR/20} + 1} \approx 24 \% \quad (5.54)$$

Sustituyendo $r=24$ en la expresión (5.53) se concluye que son necesarias 135497 muestras o un tiempo de 7.45 segundos, para obtener la separación con $\varepsilon < 0.02$ suponiendo un nivel del ruido del 24% ($SNR=10$ dBs).

Como conclusión del análisis efectuado en esta sección, puede afirmarse:

1. Si el procedimiento propuesto en este capítulo utiliza $m=200.000$ muestras ($t \approx 9$ segundos) pueden separarse mezclas corrompidas con ruido uniforme (relación señal/ruido del orden de $SNR=7.5$ dBs a 23 dBs), dependiendo de los valores concretos de los elementos de la matriz de mezcla (Tabla 5.12).
2. El procedimiento en cierta medida es inmune al ruido en el sentido de que se puede realizar adecuadamente la separación (sin modificación de los algoritmos) para niveles superiores de ruido, sin más que aumentar el número de muestras. Así, para la matriz A_8 se ha obtenido la relación (5.53) que proporciona el número de muestras necesarias para separar mezclas con un determinado nivel de ruido.

Es conveniente indicar que los resultados presentados en esta sección (incluidos en distintas tablas) se han obtenido con el programa de simulación descrito en el Capítulo 3, y considerando sólo 10 medidas en cada caso. Al ser las señales procesadas de tipo aleatorio, se obtendrían valores más precisos si se hubieran tomado para cada caso un número mayor de medidas. No obstante, consideramos que la metodología y resultados cualitativos son plenamente válidos.

En el Apéndice 3 se incluyen resultados análogos a los de esta sección pero para las 8 matrices consideradas.

5.8 CONCLUSIONES

En este capítulo se ha presentado un procedimiento adaptativo para la recuperación en tiempo real de señales cuyas funciones de distribución de probabilidad sean unimodales simétricas. La idea fundamental radica en que los ejes de la distribución de puntos de mezcla son paralelos a las aristas del hiperparalelepípedo del espacio de mezclas y por tanto, las pendientes de aquellos son las columnas de una matriz W similar a A .

Este procedimiento tiene la ventaja sobre el de máxima separación angular (Capítulo 4) que no es necesario conseguir puntos en las aristas del hiperparalelepípedo, obtención que con frecuencia es poco probable.

Se han presentado dos algoritmos, uno en línea y otro fuera de línea, y se han medido las prestaciones del algoritmo con diferentes matrices de mezcla, tanto con señales sintéticas como reales. El algoritmo se considera especialmente útil para separar señales de audio, habiéndose probado con diversas fuentes de discurso continuo y de música.

También se ha estudiado la precisión y convergencia del procedimiento en función del número de clases o vectores que se consideran para estimar los ejes de la distribución, los valores de la matriz de mezcla (el procedimiento no es equivariante), y el nivel de ruido introducido en las mezclas.

CAPÍTULO 6:

CONCLUSIONES, APORTACIONES Y LÍNEAS FUTURAS DE INVESTIGACIÓN

En esta memoria se han presentado nuevos algoritmos para la separación ciega de señales, basados en propiedades geométricas de dichas señales y de la transformación producida en el proceso lineal de mezcla.

Las principales aportaciones y conclusiones obtenidas pueden resumirse en los siguientes puntos:

- 1) Se ha realizado una revisión de los procedimientos de separación de señales más relevantes propuestos hasta la fecha, fundamentados en un enfoque estadístico del problema y que parten de la hipótesis de la independencia estadística de las fuentes. En vez de presentar aisladamente los procedimientos indicados, se ha efectuado una abstracción tanto de los principios en que se basan como de las arquitecturas que los implementan,

buscando una unidad conceptual a partir de la cual entender mejor los diversos métodos de separación. Una vez establecido ese marco general y coherente, se resumen los métodos de separación más conocidos, basados en: la minimización de la dependencia estadística (Análisis de Componentes Independientes, ICA), la estimación de máxima verosimilitud, el principio de máxima entropía, medidas estadísticas de alto orden, y en el análisis de componentes principales no-lineal (PCA no lineal). (*Capítulo 1*).

- 2) Se clasifican e identifican los distintos campos de aplicación de la separación ciega de fuentes. (*Capítulo 1*).
- 3) Se perfilan diversas aportaciones que versan sobre modelos de mezcla más complejos que el lineal instantáneo, y otros temas colaterales que definen distintas líneas de investigación abiertas en el campo de la separación ciega de fuentes (*Capítulo 1*).
- 4) Se formalizan las hipótesis y propiedades en las que se basan los métodos geométricos de separación ciega de señales, exponiéndolas de forma sistemática y completa. La hipótesis fundamental de los métodos geométricos de separación es que las fuentes están acotadas, y, como consecuencia de ello, la matriz de mezcla puede estimarse conociendo las pendientes de las aristas del hiperparalelepípedo que contiene a los vectores de mezcla. (*Capítulo 2*).
- 5) Se ha diseñado e implementado un entorno de simulación para facilitar el desarrollo de algoritmos de separación ciega de fuentes. La herramienta dispone de una biblioteca de señales, tanto sintéticas como reales, y permite simular la mezcla eligiendo los elementos de la matriz de mezcla, bien arbitrariamente por el usuario, o bien aleatoriamente. El conjunto de programas incluye algoritmos para la estimación de la matriz de mezcla a partir de las observaciones, y procedimientos para recuperar las señales originales conociendo las señales de mezcla y la matriz estimada. El simulador también implementa distintos índices para medir las prestaciones de los resultados obtenidos: error absoluto, RMS entre señales originales y recuperadas, RMS entre matriz de mezcla y matriz estimada, diafonía e índice de prestaciones de Amari. Por otra parte, en cualquier

momento, entre las distintas fases de simulación, las distintas señales e índices involucrados en el proceso pueden visualizarse en forma de frecuencia de amplitudes, evolución en el tiempo y vectorialmente. Los programas se han desarrollado en MatLab, la interfaz con el usuario es interactiva y basada en menús, y la modularidad del sistema completo permite ampliarlo con facilidad, incluyendo nuevas funciones y algoritmos.

(Capítulo 3).

- 6) Se ha desarrollado un algoritmo geométrico original para separación ciega de señales. Este algoritmo, a partir de los vectores de observación, estima uno de los vértices del hiperparalelepípedo, que se toma como origen, y un conjunto de p vectores, cada uno de ellos perteneciente a una arista distinta de las que inciden en dicho vértice. Esta estimación se efectúa considerando que el conjunto de p vectores buscados, satisfacen la condición de maximizar su separación angular. El algoritmo se implementa con dos redes neuronales artificiales con pesos comunes, una recursiva para recuperar las señales originales, y otra con aprendizaje competitivo que obtiene los valores de los pesos. El algoritmo se prueba con el entorno de simulación citado anteriormente, utilizando señales tanto sintéticas como reales (palabras aisladas), mostrándose las prestaciones y limitaciones del mismo. Con señales de voz se obtiene una recuperación de las señales originales en menos de 2 segundos. Con señales sintéticas a las que se le añade un ruido blanco acotado con una SNR de 8 dBs, se obtuvo un error cuadrático medio entre los coeficientes de la matriz estimada y la matriz de mezcla menor de 0.015, tras haber procesado 17.000 muestras. Se muestra también que el procedimiento se puede aplicar a señales estadísticamente dependientes, siempre que tal dependencia no implique la no generación de p vectores próximos a las aristas del hiperparalelepípedo que convergen en el vértice considerado como origen. *(Capítulo 4).*

- 7) Se presenta un nuevo algoritmo que no tiene la limitación del anteriormente descrito, en el sentido de que es aplicable a fuentes con funciones de distribución de probabilidad unimodales y centradas; es decir, la probabilidad de que se generen vectores en las aristas del hiperparalelepípedo puede ser muy baja, e incluso nula. Para señales con función de densidad de probabilidad con las características indicadas (unimodales y centradas), se

observa que en la distribución de puntos en el espacio de mezclas existen direcciones (que denominamos *ejes de la distribución de vectores de mezcla*) en las que la densidad de puntos es notablemente mayor. Estos ejes de la distribución pasan por el centro del hiperparalelepipedo y son paralelos a sus aristas. Como consecuencia de ello, conociendo las pendientes de los ejes de la distribución se puede estimar la matriz de mezcla. El procedimiento se implementa efectuando una traslación de los vectores de mezcla tomando como nuevo origen el centro del hiperparalelepipedo, y clasificando los vectores de mezcla de acuerdo con el ángulo que forman con la horizontal. Las clases con mayor frecuencia de incidencia de vectores de mezcla, corresponden a las direcciones de los ejes de la distribución. Se prueba y miden las prestaciones del nuevo algoritmo con ayuda del entorno de simulación desarrollado, utilizando tanto señales sintéticas como reales (voz y música) obteniéndose resultados satisfactorios. En todos los casos se obtiene una separación de señales de voz y voz+música adecuada en tiempos del orden de 2 segundos (unas 50000 muestras). También se efectúa un análisis de los efectos del ruido, así como de la convergencia del método.

- 8) Se incluyen dos apéndices que contienen conceptos básicos y propiedades relacionadas con funciones de distribución estadísticas y la Teoría de la Información, respectivamente, que son utilizados en el presente trabajo. Estos apéndices tienen por objeto facilitar la comprensión de esta memoria, ya que así el lector tiene una referencia próxima a dichos conceptos y propiedades, y en la notación que se sigue en los distintos capítulos. (*Apéndices I y II*).

Por otra parte, el trabajo descrito en esta memoria abre nuevas perspectivas de investigación y desarrollo. Algunos temas que se pueden acometer de forma inmediata, y que quedaron fuera del alcance inicialmente propuesto para este trabajo, son:

- a) Incluir en el entorno de simulación implementado, algoritmos desarrollados por otros autores y que aparecen en la literatura bien consolidados. De esta forma el entorno de simulación podrá utilizarse como un auténtico banco de prueba en el que se podrán comparar de forma objetiva distintos procedimientos de separación ciega de señales.
- b) Utilizar en nuevas aplicaciones los algoritmos desarrollados, concretamente en aplicaciones de imágenes y de biomedicina.
- c) Como uno de los puntos más rápidos y fáciles de estimar es el centro del hiperparalelepipedo, se propone, de acuerdo con los desarrollos teóricos efectuados en el Capítulo 2, explorar las posibilidades de modificar el algoritmo de máxima separación angular, de forma que se utilice el centro para obtener con mayor precisión y rapidez el vértice de cambio de coordenadas de los vectores de mezcla.
- d) Ampliar el algoritmo de separación basado en la determinación de los ejes de la distribución de vectores de mezcla, para el caso de más de dos fuentes.
- e) Los algoritmos se han probado con señales reales y mezclas sintéticas. Deberían probarse, y adaptarse en su caso, para su funcionamiento adecuado con mezclas reales de señales reales, lo que implicaría la separación de señales en lo que, comúnmente, se denominan *medios convolutivos*.

A más largo plazo se consideran de gran interés los siguientes tres temas:

- a) Obtener modelos adecuados para mezclas obtenidas en aplicaciones reales.
- b) Ampliar los algoritmos geométricos a modelos de mezcla reales (convolutivo y no-lineal).
- c) Estudiar las conexiones entre los métodos basados en el enfoque geométrico y el enfoque estadístico.

Consideramos, como principal conclusión de este trabajo, que los métodos de separación ciega de señales basados en conceptos geométricos constituyen un nuevo enfoque que da lugar a algoritmos fáciles de comprender e implementar, con tiempos de convergencia adecuados, y que además pueden superar algunas de las limitaciones que presentan los basados en principios de naturaleza estadística.



APÉNDICE 1:

CONCEPTOS BÁSICOS DE ESTADÍSTICA UTILIZADOS EN LOS MÉTODOS DE SEPARACIÓN DE FUENTES.

En este apéndice se incluyen algunos conceptos básicos de estadística que ayudarán a comprender mejor los contenidos de la presente memoria ya que con ellos se tiene una referencia próxima y en la notación que se usa a lo largo de la memoria. Se describen sólo aquellas funciones de densidad de probabilidad que se citan en los Capítulos 1, 4 y 5, aunque los métodos propuestos pueden aplicarse a otros tipos de señales. Para más detalles pueden consultarse, por ejemplo, las referencias bibliográficas [JOH95, JOH94a, JOH94b, LAC97, MEN95, STU94].

A1.1 INTRODUCCIÓN

En el análisis de una señal aleatoria, s , es de gran interés el estudio de su función de distribución, $f(s)$ ($s_m \leq s \leq s_M$), ya que ésta proporciona la probabilidad de que se genere un valor determinado, s , de la señal dentro del rango en que esté definida la señal, $s \in [s_m, s_M]$.

En la práctica podemos aproximar la distribución de probabilidad de una señal a partir de su distribución de frecuencias (número de veces que se repiten los elementos de cada una de las clases en que se subdivide el rango de valores de la variable aleatoria), sin más que dividir el número de elementos de cada clase por el número total de muestras, NM.

Un conjunto de NM observaciones de una señal, $s(1), s(2), \dots, s(NM)$, puede describirse, además de por su función de distribución, por medio de unas medidas descriptivas que especifican, entre otras propiedades la *localidad* o *tendencia central* (media, mediana, moda, etc.), la *dispersión* o *variación* (rango, desviación media, desviación estándar, varianza, etc.), la *asimetría* ("skewness") y la *agudeza* ("kurtosis").

A1.2 MOMENTOS Y CUMULANTES

A1.2.1 Momentos

Los *momentos* de una variable aleatoria pueden utilizarse para describir su distribución de probabilidad [MEN95, STU94]. El k -ésimo momento, μ'_k , de la variable aleatoria s alrededor del origen (0) se define como:

$$\mu'_k(s) = E\{s^k\} = \int_{-\infty}^{\infty} s^k \cdot f(s) \cdot ds \quad k = 1, 2, \dots \quad (\text{A1.1})$$

donde $E\{\cdot\}$ es la esperanza matemática:

$$E\{s\} = \int_{-\infty}^{\infty} s \cdot f(s) \cdot ds \quad (\text{A1.2})$$

El k-ésimo momento, μ_k , de la variable s alrededor de su *media*, $\mu' = \langle s \rangle$, se define como:

$$\mu_k(s) = E\{(s - \mu)^k\} = \int_{-\infty}^{\infty} (s - \mu)^k \cdot f(s) \cdot ds \quad (\text{A1.3})$$

Dos de los momentos más utilizados son la *media* $\mu = \mu'$, y la *varianza*, $\sigma^2 = \mu_2$, donde σ es la *desviación estándar*. Si c es una constante y s una variable aleatoria con media μ y varianza σ^2 , se verifican las siguientes propiedades [MEN95]:

$$E\{c\} = c \quad (\text{A1.4})$$

$$E\{c \cdot s\} = c \cdot E\{s\} \quad (\text{A1.5})$$

$$\sigma^2 = E\{(s - \mu)^2\} = E\{s^2\} - \mu^2 \quad (\text{A1.6})$$

$$\sigma_{(c+s)}^2 = \sigma^2 \quad (\text{A1.7})$$

$$\sigma_{c \cdot s}^2 = c^2 \cdot \sigma^2 \quad (\text{A1.8})$$

Los momentos de una variable aleatoria se pueden estimar directamente, a partir de las definiciones (A1.1) y (A1.3) o a partir de la *función generadora de momentos*, $m(t)$, de dicha variable que se define como:

$$m(t) = E\{e^{ts}\} \quad (\text{A1.9})$$

Si $m(t)$ existe, los distintos momentos alrededor del origen se pueden calcular con la expresión:

$$\mu_k = \left. \frac{d^k m(t)}{dt^k} \right]_{t=0} \quad (\text{A1.10})$$

Bajo ciertas condiciones, que suelen cumplir la mayoría de las funciones de distribución, si dos variables aleatorias, s_1, s_2 , tienen sus respectivos momentos iguales:

$$\mu_k(s_1) = \mu_k(s_2) \quad k = 1, 2, \dots \quad (\text{A1.11})$$

sus distribuciones de probabilidad, $f_1(s_1)$ y $f_2(s_2)$, son idénticas [MEN98], [STU94]. De hecho, con que tengan iguales los 3 o 4 primeros momentos, ya pueden considerarse idénticas.

A1.2.2 Cumulantes

Los *cumulantes*, κ_k , son otras constantes que describen y especifican propiedades y circunstancias de una distribución y son más útiles, desde un punto de vista teórico, que los momentos. Los cumulantes de orden k pueden calcularse a partir de los momentos de orden igual o inferior a k . Los cumulantes de orden 1 a 4, respecto al origen y en función de los momentos son [STU94]:

$$\begin{aligned}\kappa_1 &= \mu_1' \\ \kappa_2 &= \mu_2' - \mu_1'^2 \\ \kappa_3 &= \mu_3' - 3\mu_2' \cdot \mu_1' + 2\mu_1'^3 \\ \kappa_4 &= \mu_4' - 4\mu_3' \cdot \mu_1' - 3\mu_2'^2 + 12\mu_2' \mu_1'^2 - 6\mu_1'^4\end{aligned}\tag{A1.12}$$

Si las variables aleatorias están centradas en su media, es decir, $\mu_1' = 0$, las expresiones de los cumulantes se simplifican [STU94]:

$$\begin{aligned}\kappa_1 &= 0 \\ \kappa_2 &= E\{s^2\} = \mu_2 \\ \kappa_3 &= E\{s^3\} = \mu_3 \\ \kappa_4 &= E\{s^4\} - 3E\{s^2\}^2 = \mu_4 - 3\mu_2^2\end{aligned}\tag{A1.13}$$

Para una variable gaussiana (Sec.A1.3.2) los cumulantes de orden superior a 2 son todos nulos; es decir, $\kappa_k = 0$ ($\forall k > 2$).

La varianza, σ^2 , cumulante de s de orden 2, $\sigma^2 = \kappa_2$, caracteriza la *potencia* o *dispersión* de s .

Centrando en la media y normalizando la señal aleatoria, es decir, efectuando el siguiente cambio de variable:

$$\tilde{s} = \frac{s - \mu}{\sqrt{\kappa_2}} = \frac{s - \mu}{\sigma} \quad (\text{A1.14})$$

se obtiene la variable o *señal estandarizada*, \tilde{s} ; es decir, toda señal estandarizada tiene media 0 y varianza 1. Los *cumulantes estandarizados*, $\tilde{\kappa}_k$, son los cumulantés de la señal estandarizada.

El cumulante estandarizado de orden 3, $\tilde{\kappa}_3$, mide la *asimetría* ("skewness") de la función de distribución ya que es nula si la función de densidad de probabilidad de s posee un eje de simetría. Si s es centrada, el *factor de asimetría* (γ_1) resulta ser:

$$\gamma_1 = \tilde{\chi}_3 = \frac{E\{s^3\}}{E\{s^2\}^{3/2}} = \frac{\mu_3}{\mu_2^{3/2}} = \frac{\kappa_3}{\kappa_2^{3/2}} \quad (\text{A1.15})$$

Si la distribución es simétrica: $\gamma_1=0$ (ya que toda distribución de probabilidad simétrica tiene los momentos impares nulos, y en particular $\mu_3=0$). Usualmente si $\gamma_1>0$ quiere decir que la cola superior de la distribución es la más baja (media > mediana > moda) y si $\gamma_1<0$ la cola inferior es la más baja (media < mediana < moda).

El cumulante estandarizado de orden 4, $\tilde{\kappa}_4$, es el *factor de agudeza* o *kurtosis* (γ_2). Si la señal está centrada en la media:

$$\gamma_2 = \tilde{\kappa}_4 = \frac{E\{s^4\}}{E\{s^2\}^2} - 3 = \frac{\mu_4}{\mu_2^2} - 3 = \frac{\kappa_4}{\kappa_2^2} \quad (\text{A1.16})$$

El signo de la agudeza caracteriza el comportamiento en el infinito de las densidades de probabilidad unimodales. Si $\gamma_2=0$ se dice que la distribución es mesoaguda (gaussiana). Si la agudeza es negativa ($\gamma_2<0$) se dice que la distribución es *sub-gaussiana* (o *platicúrtica*), ya que tiende a cero en el infinito más rápidamente que la distribución gaussiana. Por contrario, si la agudeza es positiva ($\gamma_2>0$), la función de distribución tiende a cero en el

infinito más lentamente que la distribución gaussiana se dice que es *supra-gaussiana* (o *leptocúrtica*).

A1.2.3 Distribuciones de probabilidad multivariantes. Cumulantes cruzados

La *distribución de probabilidad conjunta* $f(\mathbf{s})=f(s_1, \dots, s_p)$ de un conjunto de variables aleatorias discretas $\mathbf{s}=(s_1, \dots, s_p)$ proporciona los valores de probabilidad $f(\mathbf{s})$ para todas las combinaciones posibles de valores (s_1, \dots, s_p) .

Por definición, la distribución de probabilidad conjunta verifica las siguientes propiedades:

$$\text{PA1.1 } 0 \leq f(s_1, \dots, s_p) \leq 1 \quad \forall s_1, \dots, s_p$$

$$\text{PA1.2 } \sum_{s_1} \sum_{s_2} \cdots \sum_{s_p} f(s_1, \dots, s_p) = 1$$

La *distribución de probabilidad marginal* (o *incondicional*) $f(s_i)$ de s_i representa la probabilidad de que se produzcan los valores de s_i independientemente de los otros valores s_j ($j \neq i$). Se verifica:

$$\begin{aligned} f_1(s_1) &= \sum_{s_2} \sum_{s_3} \cdots \sum_{s_p} f(s_1, \dots, s_p) \\ &\dots \\ f_p(s_p) &= \sum_{s_1} \cdots \sum_{s_{p-1}} f(s_1, \dots, s_p) \end{aligned} \tag{A1.17}$$

La probabilidad de que se produzca una combinación (s_1, s_2) será igual a la probabilidad de que se produzca s_2 condicionado a que se produzca simultáneamente s_1 , $f(s_1|s_2)$, multiplicado por la probabilidad de que se produzca s_1 . Este hecho matemáticamente se puede expresar así:

$$f(s_1, s_2) = f(s_1|s_2) \cdot f(s_1) \tag{A1.18}$$

Al término $f(s_1|s_2)$ se le denomina *probabilidad condicional* (de s_2 con respecto a s_1).

Análogamente podemos decir:

$$f(s_1, s_2) = f(s_2|s_1) \cdot f(s_2) \quad (\text{A1.19})$$

Se verifica que:

$$f(s_1|s_2) \leq f(s_1); \quad f(s_2|s_1) \leq f(s_2); \quad (\text{A1.20})$$

es decir, la probabilidad de que se produzca s_1 habiéndose producido s_2 es menor o igual a la probabilidad de que aisladamente se produzca s_1 (o la probabilidad de que se produzca s_2 habiéndose producida s_1 es menor o igual que la probabilidad de que se produzca s_2). La igualdad se obtendrá caso de que la generación de s_1 no dependa de s_2 (o que la generación de s_2 no dependa de s_1); es decir, cuando s_1 y s_2 sean estadísticamente independientes.

De (A1.18), (A1.19) y (A1.20) se deduce que:

$$f(s_1, s_2) \leq f(s_1) \cdot f(s_2) \quad (\text{A1.21})$$

verificándose la igualdad cuando s_1 y s_2 sean estadísticamente independientes. En este caso la igualdad es obvia ya que indica que la probabilidad de la generación de una pareja (s_1, s_2) depende de que se produzca el valor s_1 y aisladamente el valor s_2 .

En general, p señales s_1, \dots, s_p son estadísticamente independientes si y sólo si:

$$f(s_1, \dots, s_p) = \prod_{i=1}^p f(s_i) \quad (\text{A1.22})$$

Las mismas definiciones y conclusiones se pueden aplicar a variables aleatorias continuas sin más que sustituir en PA2.1 y en (A1.17) las sumatorias por integrales.

Supóngase una variable aleatoria p -dimensional, $s^T = (s_1, \dots, s_p)$ que está centrada en su media; es decir $E\{s_i\} = 0$ para todo i . Del mismo modo que en el caso escalar (A1.9) se puede definir una función característica conjunta, de p variables aleatorias, s_i , como:

$$m(\mathbf{v}) \hat{=} E \left\{ e^{j \sum_i v_i s_i} \right\} = E \left\{ e^{j \mathbf{v}^T \mathbf{x}} \right\} \quad (\text{A1.23})$$

donde $\mathbf{v}^T = (v_i)$. Si las componentes s_i del vector aleatorio \mathbf{s} admiten una densidad de probabilidad conjunta $f(\mathbf{s})$ entonces la función generadora de momentos de \mathbf{s} viene dada por la transformada de Fourier de dicha densidad:

$$m(\mathbf{v}) = \int_{\mathbb{R}^p} e^{j\mathbf{v}^T \mathbf{u}} \cdot f_{\mathbf{s}}(\mathbf{u}) \cdot d\mathbf{u} \quad (\text{A1.24})$$

Los cumulantes de orden 2, o covarianzas, son magnitudes con dos subíndices que se pueden organizar en una *matriz de covarianza* cuyos elementos son de la forma:

$$\begin{aligned} \text{Cum}[s_i, s_j] &= \kappa_{s_{ij}} = \kappa_{s,2} = E\{\mu_1(s_1)\mu_1(s_2)\} = \\ &= E\{s_1, s_2\} - \mu_1(s_1)\mu_1(s_2) = \mu_1(s_1, s_2) - \mu_1(s_1)\mu_1(s_2) \end{aligned} \quad (\text{A1.25})$$

La expresión anterior indica diversas formas de representar los cumulantes (en este caso de orden 2). Se suele utilizar una u otra forma dependiendo del contexto.

Los cumulantes de orden 3 son:

$$\text{Cum}[s_1, s_2, s_3] = \kappa_{s,3} \quad (\text{A1.26})$$

Los cumulantes de orden 4 son:

$$\begin{aligned} \text{Cum}[s_1, s_2, s_3, s_4] &= \kappa_{s,4} = E\{\mu_1(s_1)\mu_1(s_2)\mu_1(s_3)\mu_1(s_4)\} - \\ &\quad - E\{\mu_1(s_1)\mu_1(s_2)\}E\{\mu_1(s_3)\mu_1(s_4)\} - \\ &\quad - E\{\mu_1(s_1)\mu_1(s_3)\}E\{\mu_1(s_2)\mu_1(s_4)\} - \\ &\quad - E\{\mu_1(s_1)\mu_1(s_4)\}E\{\mu_1(s_2)\mu_1(s_3)\} - \end{aligned} \quad (\text{A1.27})$$

En general, los cumulantes de orden r se relacionan con los momentos según la fórmula de Leonov y Shirayev [LAC97]:

$$\text{Cum}[s_1, \dots, s_r] = \sum (-1)^{k-1} (k-1)! E\left\{ \prod_{i \in \nu_1} s_i \right\} \cdot E\left\{ \prod_{j \in \nu_2} s_j \right\} \cdots E\left\{ \prod_{k \in \nu_p} s_k \right\} \quad (\text{A1.28})$$

donde la sumatoria se extiende a todos los conjuntos $\{v_1, \dots, v_p\}$, $1 \leq p \leq r$ que forman una partición de $\{1, \dots, r\}$, y k es el número de elementos que componen la partición.

Se denomina *orden de la estadística* al número de índices, r , utilizados para representar los cumulantes.

Una propiedad de gran importancia utilizada en diversos métodos de separación ciega de señales indica que varias señales aleatorias son estadísticamente independientes si y sólo si sus cumulantes cruzados de cualquier orden son nulos [LAC97]. Es conveniente hacer notar que la anulación de los cumulantes cruzados de orden 2 (covarianzas) es condición necesaria pero no suficiente (salvo en el caso de distribuciones gaussianas) para la independencia estadística, ya que lo deben ser también los de orden superior. Cuando las covarianzas entre señales son nulas, se dice que las señales están *decorrelacionadas*.

A1.3 FUNCIONES DE DENSIDAD DE PROBABILIDAD DE SEÑALES ALEATORIAS REFERIDAS EN LA PRESENTE MEMORIA.

A lo largo de esta memoria se hace uso de diversas funciones de distribución, que para comodidad del lector, se definen a continuación incluyendo las propiedades de mayor interés dentro de la perspectiva de los nuevos métodos de separación de señales que se introducen.

A1.3.1 Señales con densidad de probabilidad uniforme.

Una señal aleatoria se dice que tiene una densidad de probabilidad uniforme en el rango $[s_m, s_M]$ si todo valor de la señal comprendido en ese rango tiene igual probabilidad de producirse (Figura A1.1).

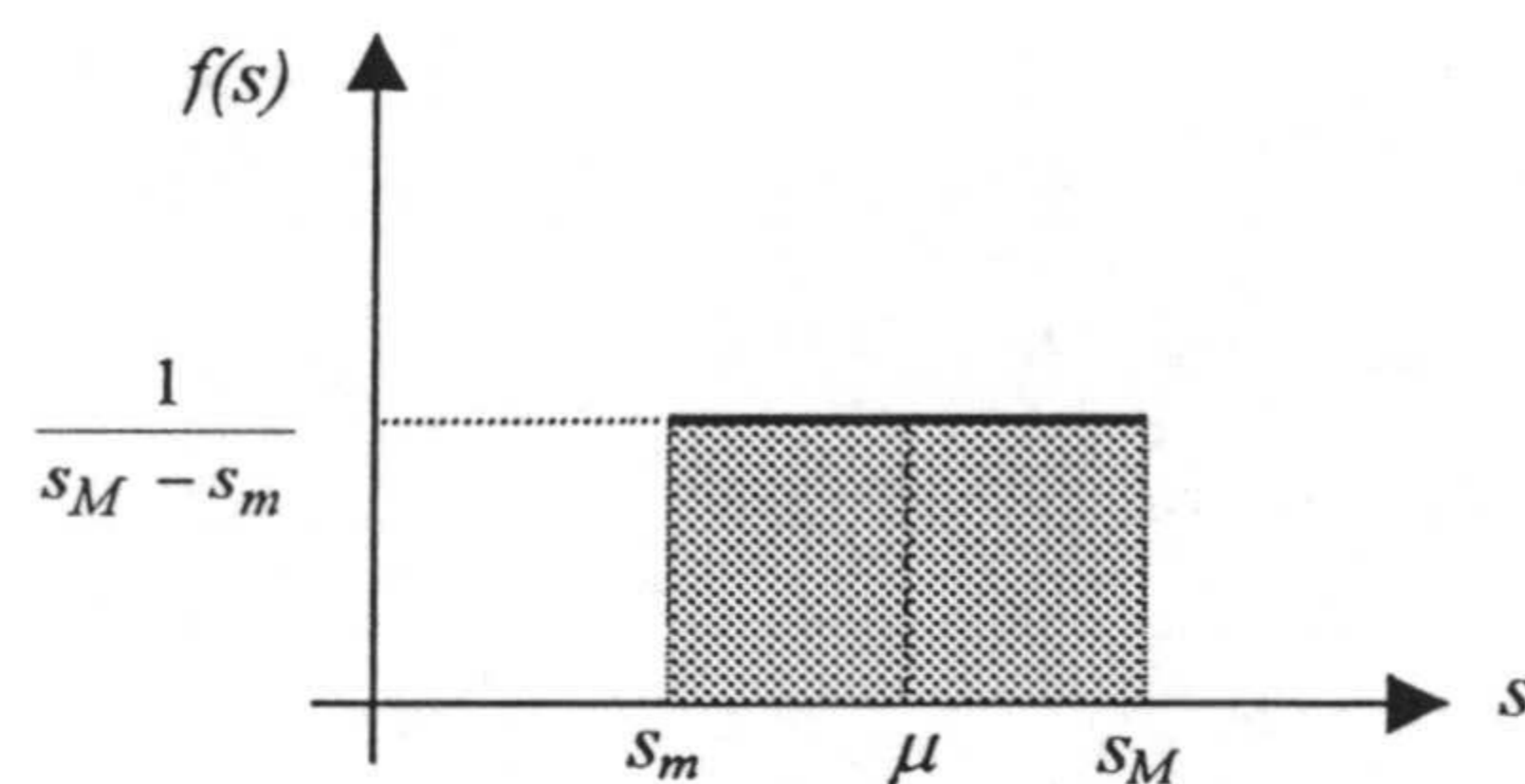


Figura A1. 1. *Función de densidad de probabilidad uniforme.*

Teniendo en cuenta que el área total bajo la curva de probabilidad (área sombreada de la Figura A1.1) tiene que ser la unidad, se verifica que:

$$\begin{cases} f(s) = \frac{1}{s_M - s_m} & \text{si } s_m \leq s \leq s_M \\ f(s) = 0 & \text{en otro caso} \end{cases} \quad (\text{A1.29})$$

La función generadora de momentos de una distribución de probabilidad uniforme es:

$$m(t) = \frac{e^{ts_M} - e^{ts_m}}{t(s_M - s_m)} \quad (\text{A1.30})$$

Los momentos de orden par son:

$$\mu_k = \frac{1}{k+1} \left(\frac{s_M - s_m}{2} \right)^k \quad (\text{A1.31})$$

resultando ser la media y la varianza (respectivamente):

$$\begin{aligned} \mu &= \frac{s_M + s_m}{2} \\ \sigma^2 &= \frac{(s_M - s_m)^2}{12} \end{aligned} \quad (\text{A1.32})$$

y el factor de agudeza (kurtosis):

$$\gamma_2 = -\frac{6}{5} \quad (\text{A1.33})$$

Por analogía con la función de distribución gaussiana se dice que una función de distribución es supra-uniforme si $f(s_m) > \mu$ y $f(s_M) > \mu$ sub-uniforme si $f(s_m) < \mu$ y $f(s_M) < \mu$.

A1.3.2 Señal con densidad de probabilidad gaussiana (normal).

La función de distribución gaussiana también se denomina *distribución normal* debido a que es el modelo al que se ajustan multitud de distribuciones que aparecen en muchos campos científicos y técnicos. Por ejemplo, la distribución de errores relativos en medidas experimentales corresponde a una densidad de probabilidad gaussiana.

La función de distribución gaussiana viene dada por:

$$f(s) = \frac{1}{\sigma\sqrt{2\pi}} \cdot e^{-(s-\mu)^2/(2\sigma^2)} \quad -\infty < s < \infty \quad (\text{A1.34})$$

La función generadora de los momentos es [LAC97, MEN95]:

$$m(t) = e^{\mu t + (t^2 \sigma^2 / 2)} \quad (\text{A1.35})$$

Debido a la simetría de esta función (ver Figura A1.2), los momentos de orden impar son nulos y los de orden par vienen dados por [STU94]:

$$\mu_{2r} = \sigma^{2r} \cdot \frac{(2 \cdot r)!}{r! \cdot 2^r} \quad r = 1, 2, 3, \dots \quad (\text{A1.36})$$

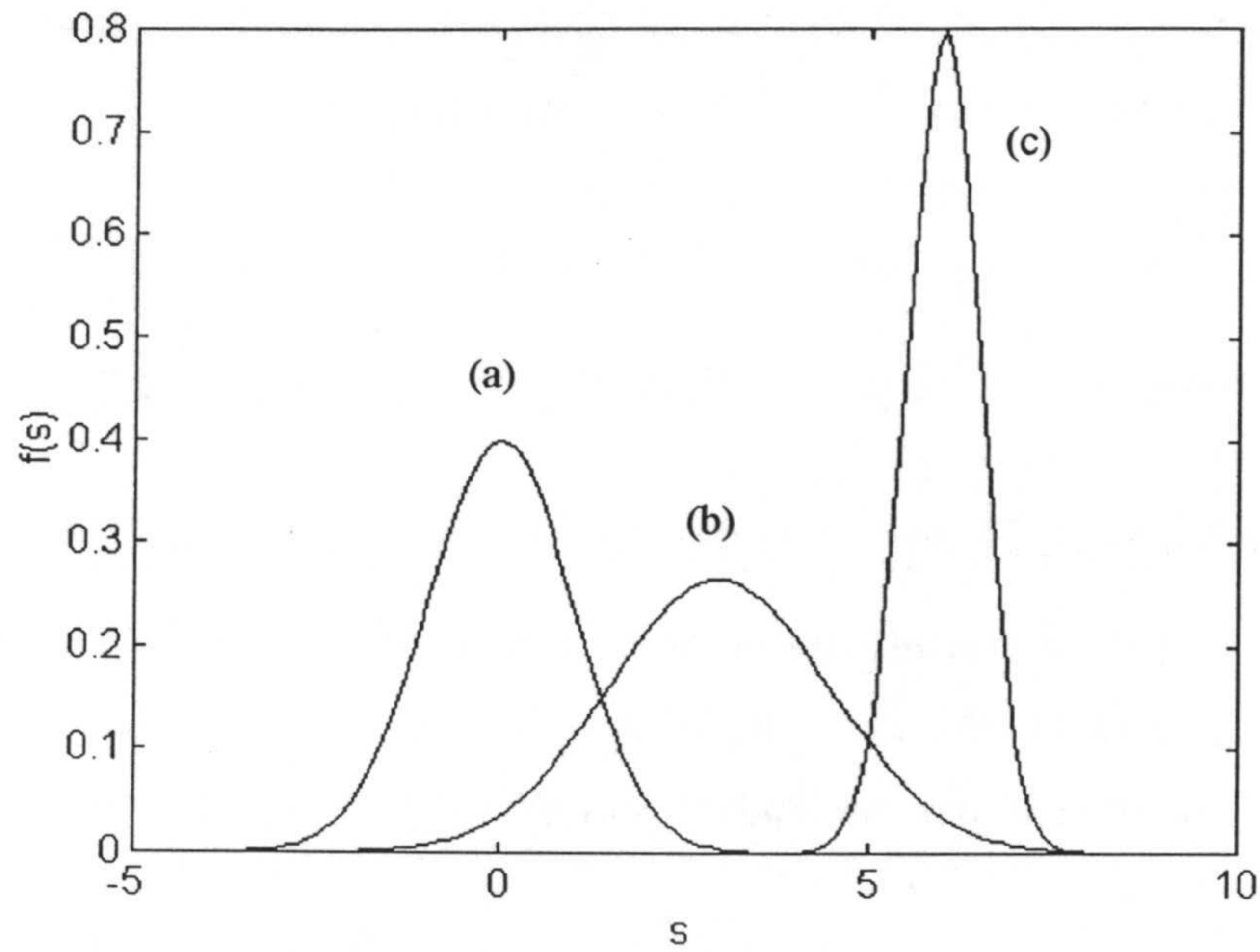


Figura A1. 2. Distribuciones de probabilidad gaussianas con:
 (a) $\mu=0, \sigma=1$; (b) $\mu=3, \sigma=1.5$; (c) $\mu=6, \sigma=0.5$.

A1.3.3 Función de densidad de probabilidad gamma.

Esta función se define como [JOH94b]:

$$f(s) = (s-\gamma)^{\alpha-1} \cdot \frac{e^{-(s-\gamma)/\beta}}{\beta^\alpha \cdot \Gamma(\alpha)} \quad \text{con } \alpha > 0, \beta > 0, s > \gamma \quad (\text{A1.37})$$

donde $\Gamma(\alpha)$ es la función gamma:

$$\Gamma(\alpha) = \int_0^\infty s^{\alpha-1} \cdot e^{-s} \cdot ds \quad (\text{A1.38})$$

El parámetro α es el *factor de forma* y β es el *factor de dispersión*. Usualmente se considera $\gamma=0$, con lo que (A1.25) se convierte en:

$$\begin{cases} f(s) = \frac{1}{\beta^\alpha \cdot \Gamma(\alpha)} \cdot s^{\alpha-1} \cdot e^{-s/\beta} & \text{con } \alpha > 0, \beta > 0, 0 \leq s < \infty \\ f(s) = 0 & \text{con } s < 0 \end{cases} \quad (\text{A1.39})$$

En la Figura A1.3 se muestran algunas funciones de probabilidad gamma.

La función generadora de momentos resulta ser [MEN95]:

$$m(t) = (1 - \beta \cdot t)^{-\alpha} \quad (\text{A1.40})$$

De forma que los momentos respecto al origen son [STU94]:

$$\begin{aligned} \mu &= \mu_1' = \alpha \cdot \beta \quad (\text{media}) \\ \mu_2' &= \alpha \cdot (\alpha + 1) \cdot \beta^2 \\ \mu_3' &= \alpha \cdot (\alpha + 1) \cdot (\alpha + 2) \cdot \beta^3 \\ \mu_4' &= \alpha \cdot (\alpha + 1) \cdot (\alpha + 2) \cdot (\alpha + 3) \beta^4 \end{aligned} \quad (\text{A1.41})$$

los momentos centrales resultan ser [STU94]:

$$\begin{aligned} \sigma^2 &= \mu_2 = \alpha \cdot \beta^2 \quad (\text{varianza}) \\ \mu_3 &= 2 \cdot \alpha \cdot \beta^3 \\ \mu_4 &= 3 \cdot \alpha \cdot (\alpha + 2) \beta^4 \end{aligned} \quad (\text{A1.42})$$

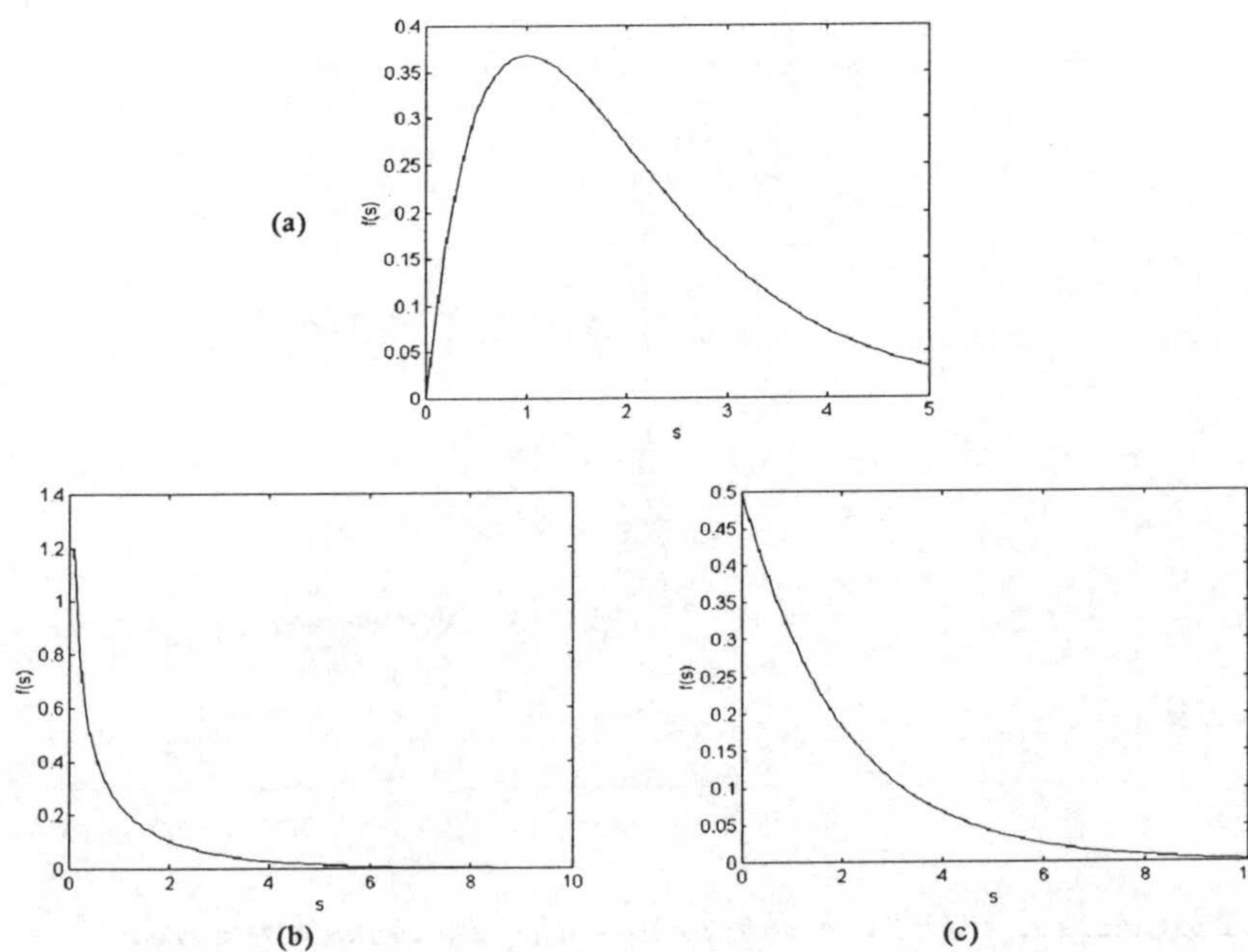


Figura A1. 3. *Distribuciones de probabilidad gamma con:*

(a) $\beta=1, \alpha=2$; (b) $\beta=2, \alpha=0.5$; (c) $\beta=2, \alpha=1$ (exponencial).

La función de densidad de probabilidad gamma es de un gran interés ya que las señales de voz se aproximan notablemente a un tipo de estas funciones que se obtiene haciendo $\alpha = \frac{1}{2}$ (Secc. , [RAB78]), con lo que teniendo en cuenta que $\Gamma(1/2) = \pi^{1/2}$, la expresión (A1.27) se convierte en:

$$f(s) = \left(\frac{1}{\beta \cdot \pi \cdot s} \right)^{1/2} \cdot e^{-s/\beta} \quad 0 \leq s < \infty \quad (\text{A1.43})$$

La densidad de probabilidad de la señal de voz, a diferencia de la dada por (A1.31, Figura A1.3b) es simétrica (bilateral), según se muestra en la Figura A1.4, con lo que para que se siga conservando que el área bajo la función de densidad de probabilidad ($-\infty < s < \infty$) sea la unidad, (A1.31) debe convertirse en:

$$f(s) = \frac{1}{2} \cdot \left(\frac{1}{\beta \cdot \pi \cdot |s|} \right)^{1/2} \cdot e^{-|s|/\beta} \quad -\infty \leq s < \infty \quad (\text{A1.44})$$

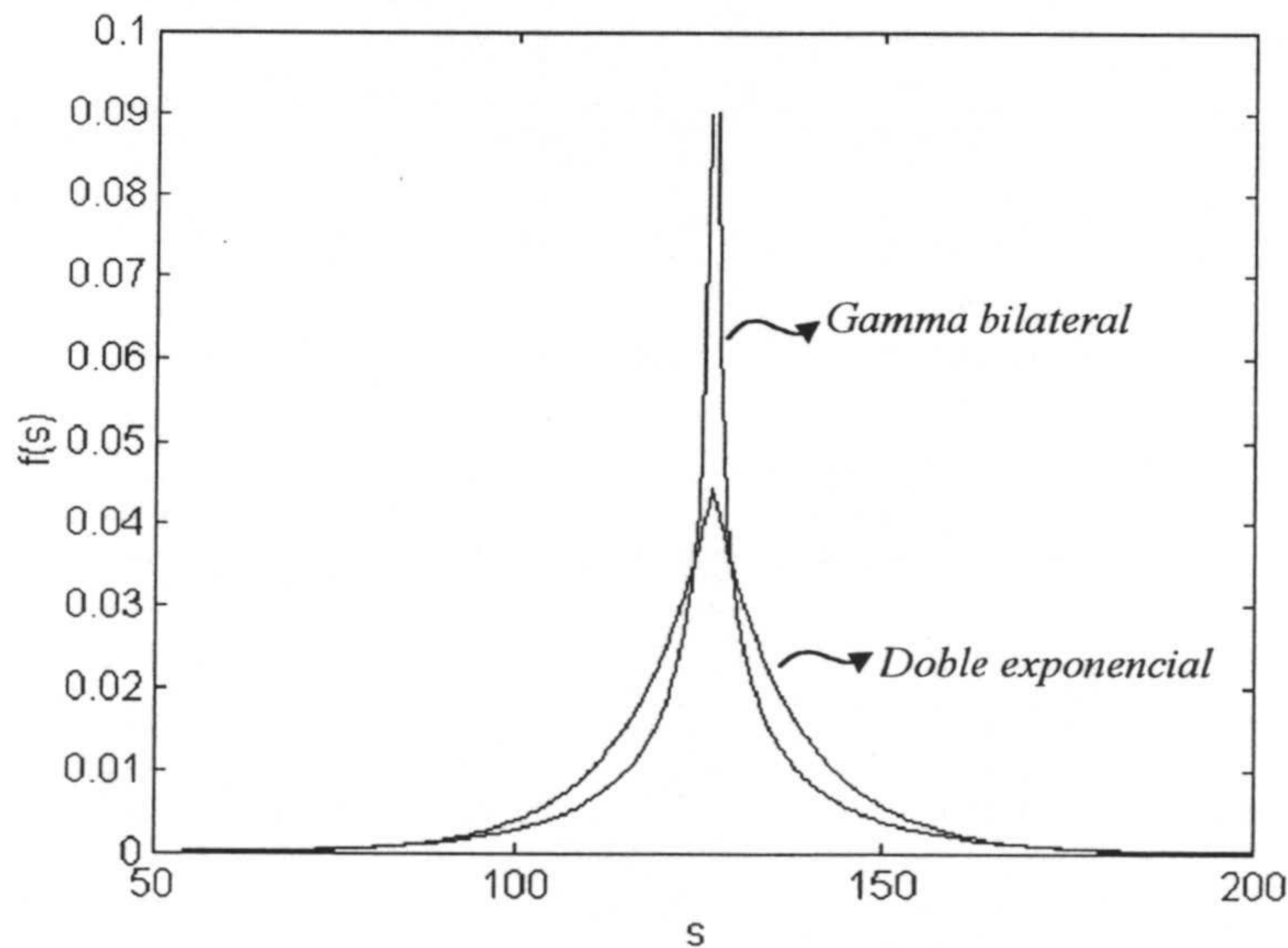


Figura A1. 4. Densidad de probabilidad gamma bilateral y doble exponencial con $\mu=127$ y $\sigma=16$.

La varianza de la función (A1.32) resulta ser:

$$\sigma^2 = \frac{3}{4} \cdot \beta^2 \quad (\text{A1.45})$$

con lo que la densidad de probabilidad gamma bilateral en función de la media (μ) y de la varianza (σ^2) es:

$$f(s) = \left(\frac{\sqrt{3}}{8 \cdot \pi \cdot \sigma \cdot |s - \mu|} \right)^{1/2} \cdot e^{-(\sqrt{3} \cdot |s - \mu| / (2\sigma))} \quad -\infty \leq s < \infty \quad (\text{A1.46})$$

La expresión anterior (A1.34) coincide con la dada por Rabiner en [RAB78].

Dos propiedades de interés son las siguientes [JOH94b]. Si s_1 y s_2 son dos variables aleatorias independientes, con función de distribución gamma y con valores del parámetro α : α_1 y α_2 , y valores de β comunes: $\beta = \beta_1 = \beta_2$, entonces se verifica:

1. $s_1 + s_2$ es una distribución gamma con igual β y con $\alpha = \alpha_1 + \alpha_2$.
2. $\frac{s_1}{s_1 + s_2}$ y $\frac{s_2}{s_1 + s_2}$ tienen distribución beta con parámetros α_1 y α_2 .

A1.3.4 Función densidad de probabilidad exponencial y doble exponencial (o de Laplace).

Si en la expresión (A1.27) de la función de probabilidad gamma se hace $\alpha=1$ se obtiene la función de probabilidad exponencial (Figura A1.3c):

$$\begin{cases} f(s) = \frac{1}{\beta} \cdot e^{-s/\beta} & \text{si } s \geq 0 \\ f(s) = 0 & \text{si } s < 0 \end{cases} \quad (\text{A1.47})$$

cuyas media, varianza, asimetría, factor de agudeza y cumulantes resultan ser respectivamente [LAC97]:

$$\mu = \beta; \quad \sigma = \beta; \quad \gamma_1 = 2; \quad \gamma_2 = 6; \quad \kappa_r = (r-1)! \sigma^r \quad (\text{A1.48})$$

Las señales de voz también se suelen aproximar con una *distribución de probabilidad doble exponencial* (también denominada *distribución de Laplace*), consistente en hacer la función exponencial (A1.35) simétrica [JOH95]:

$$f(s) = \frac{1}{2\beta} \cdot e^{-s/\beta} \quad (\text{A1.49})$$

donde ahora $\mu=0$ y $\sigma = \sqrt{2} \cdot \beta$, con lo que (A1.37) puede expresarse como:

$$f(s) = \frac{1}{\sqrt{2}\sigma} \cdot e^{-\sqrt{2} \cdot |s-\mu|/\sigma} \quad (\text{A1.50})$$

que coincide con la expresión dada por Rabiner en [RAB78].

Una forma normalizada de esta distribución se obtiene haciendo $\mu=0$ y $\sigma = \sqrt{2}$, con lo que (A1.38) se convierte en la *primera ley de error de Poisson*:

$$f(s) = \frac{1}{2} e^{-|s|} \quad (\text{A1.51})$$

El k-ésimo momento central resulta ser [JOH95]:

$$\begin{aligned} \mu_k(x) &= 0 \quad (\text{si } k \text{ es impar}) \\ \mu_k(x) &= k! \quad (\text{si } k \text{ es par}) \end{aligned} \quad (\text{A1.52})$$

y el factor de agudeza [LAC97]:

$$\gamma_2 = 3 \quad (\text{A1.53})$$

con lo que es una distribución supra-gaussiana.

En la Figura A1.4 se muestra una función de probabilidad doble exponencial superpuesta a una función gamma bilateral. La densidad de probabilidad de una señal de voz se encuentra entre ambas curvas (ver Figura 4.xx).

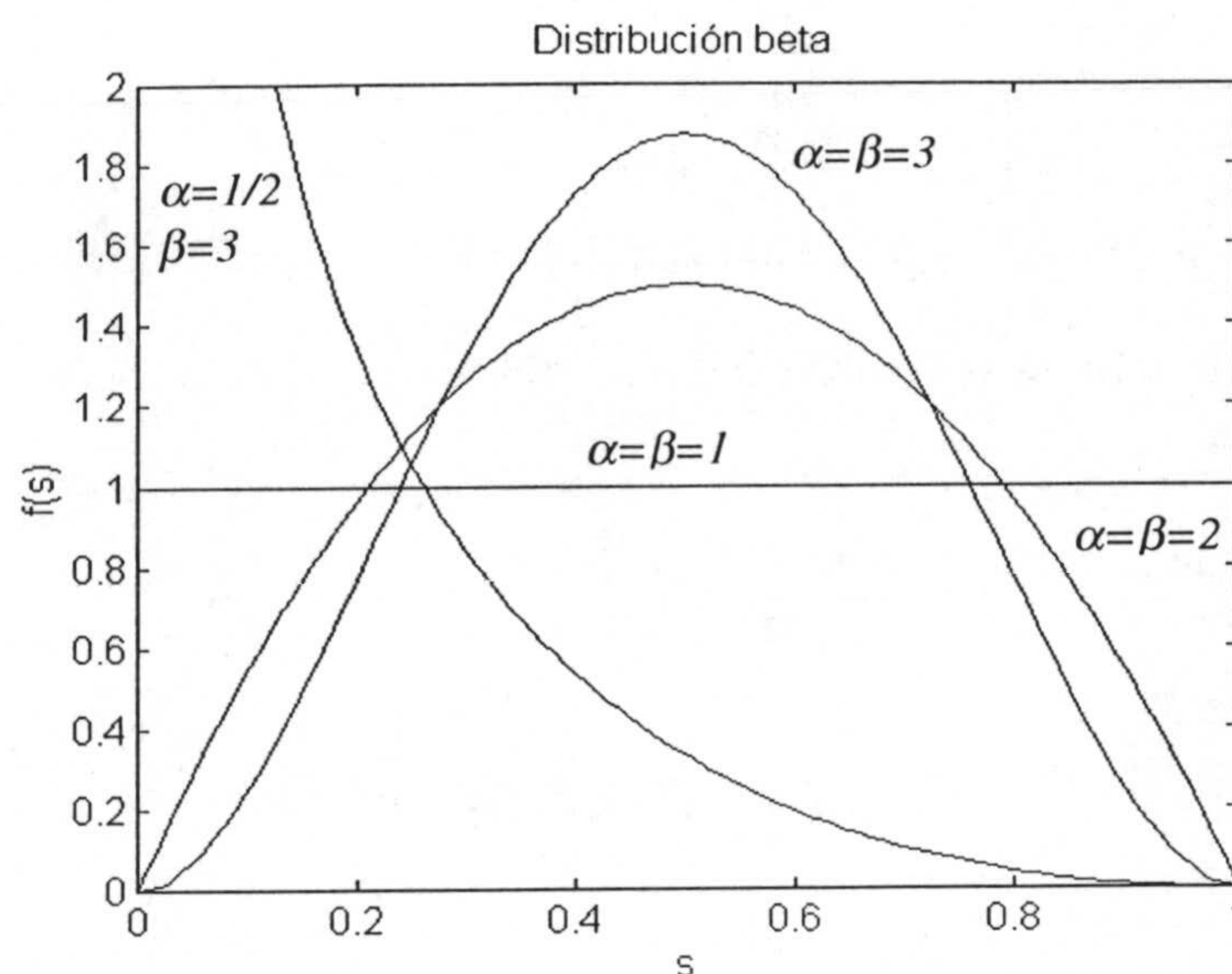


Figura A1.5. Funciones de probabilidad beta.

A1.3.5 Función densidad de probabilidad beta.

La función de densidad de probabilidad para una señal aleatoria tipo beta viene dada por [JOH95]:

$$f(s) = \frac{1}{B(\alpha, \beta)} \cdot \frac{(s - s_m)^{\alpha-1} (s_M - s)^{\beta-1}}{(s_M - s_m)^{\alpha+\beta-1}} \quad \begin{matrix} s_m \leq s \leq s_M \\ \alpha > 0, \beta > 0 \end{matrix} \quad (A1.54)$$

donde:

$$B(\alpha, \beta) = \int_0^1 s^{\alpha-1} (1-s)^{\beta-1} \cdot ds = \frac{\Gamma(\alpha) \cdot \Gamma(\beta)}{\Gamma(\alpha + \beta)} \quad (A1.55)$$

Si se hace el cambio de variable:

$$x = \frac{s - s_m}{s_M - s_m} \quad (A1.56)$$

se obtiene la forma estándar [JOH95, MEN95]:

$$f(x) = \frac{1}{B(x, \beta)} \cdot x^{\alpha-1} (1-x)^{\beta-1} \quad 0 \leq x \leq 1 \quad (A1.57)$$

Si \$\alpha=\beta\$ se obtiene una familia de *densidades de probabilidad beta simétricas* (Figura A1.5).

Si la señal está estandarizada según (A1.36), el momento r-ésimo con respecto a 0 es [JOH95]:

$$\mu'_r = \frac{\alpha^{[r]}}{[\alpha + \beta]^{[r]}} \quad (\text{A1.58})$$

donde $y^{[r]} = y(y+1)\dots(y+r-1)$ es el factorial ascendente.

En particular, la media (μ) y la varianza (σ^2) resultan ser:

$$\begin{aligned} \mu &= \frac{\alpha}{\alpha + \beta} \\ \sigma^2 &= \frac{\alpha \cdot \beta}{(\alpha + \beta)^2 \cdot (\alpha + \beta + 1)} \end{aligned} \quad (\text{A1.59})$$

La función de densidad beta proporciona un modelo adecuado para señales aleatorias, s , que tienen límites inferior y superior fijos y finitos [s_m , s_M].

APÉNDICE 2:

EL CONCEPTO DE ENTROPÍA EN EL CONTEXTO DE LA SEPARACIÓN DE FUENTES.

Una gran parte de los métodos de separación ciega de fuentes se basan en conceptos de la Teoría Estadística de la Información. Para tener una referencia próxima a estos conceptos y entender mejor los capítulos de esta memoria se incluye el presente apéndice. La notación seguida en el Capítulo 1 se corresponde con la de este apéndice y se utilizan la mayor parte de las propiedades aquí incluidas. Pueden encontrarse más detalles en las referencias [HAY99] y [LEE98], en las que se basa este apéndice.

A2.1 ENTROPÍA

De acuerdo con la terminología usada en teoría de la probabilidad, se suele utilizar una letra mayúscula para *variables aleatorias*, y la correspondiente minúscula para *el valor* de la variable aleatoria correspondiente.

Cada vez que se genere la variable aleatoria X se dice que se ha producido un *mensaje*. Estrictamente hablando, si la variable aleatoria X es continua en su rango de amplitudes, llevará una cantidad de información infinita. Sin embargo, en entornos reales (físicos o biológicos, por ejemplo) hay que reconocer que carece de sentido pensar en términos de medidas de amplitud con precisión infinita, lo que sugiere que el valor de X debe estar uniformemente cuantizado en un número finito de niveles. Es decir, X se puede interpretar como una variable aleatoria *discreta* que sigue el modelo:

$$X = \{x_k \mid k = 0, \pm 1, \dots, \pm k\} \quad (\text{A2.1})$$

donde x_k es un número discreto que representa el nivel k , y $(2k + 1)$ es el número total de niveles discretos. La separación entre niveles discretos, δx , se supone lo suficientemente pequeña para que el modelo (A2.1) represente adecuadamente a la variable considerada. También se puede pasar a la forma continua haciendo tender δx a 0 y k a infinito, en cuyo caso se tendrá una variable aleatoria continua, y habrá que transformar las sumas en integrales.

Para completar el modelo, considérese $X=x_k$ un evento con *probabilidad*:

$$p_k = P(X = x_k) \quad (\text{A2.2})$$

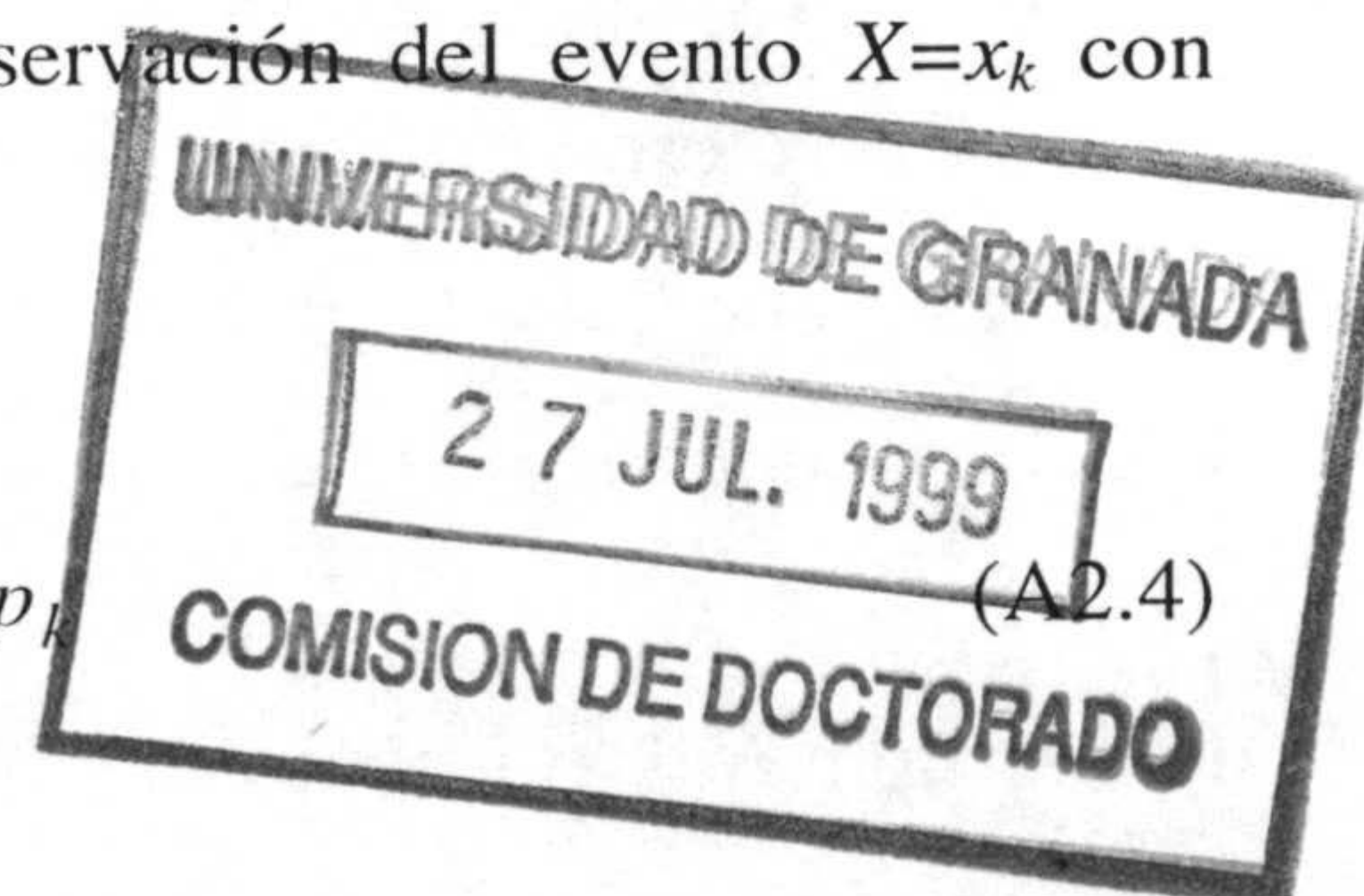
con:

$$0 \leq p_k \leq 1 \quad \text{y} \quad \sum_{k=-K}^K p_k = 1 \quad (\text{A2.3})$$

Suponiendo que el evento $X=x_k$ ocurre con probabilidad $p_k=1$, entonces $p_i=0$ para todo $i \neq k$. En esta situación no hay “sorpresas” y por tanto el evento $X=x_k$ no lleva “información” ya que se sabe a priori cual será el mensaje. Por otra parte, si los distintos niveles discretos ocurrieran con distintas probabilidades, y si en particular la probabilidad p_k fuera baja, entonces habría más “sorpresa” y por tanto, más “información” cuando X tome el valor x_k , que cuando tome otro valor x_i con mayor probabilidad p_i , $i \neq k$. Por lo tanto, las palabras “incertidumbre”, “sorpresa” e “información” están relacionadas. Antes de que ocurra el evento $X=x_k$ hay cierta incertidumbre. Cuando el evento $X=x_k$ ocurre, hay cierta sorpresa. Después de que ocurra el evento $X=x_k$ hay un incremento de información. Obviamente estas tres cantidades son iguales. Además, la cantidad de información es *inversa* a la probabilidad de ocurrencia.

Se define la cantidad de información ganada tras la observación del evento $X=x_k$ con probabilidad p_k como la siguiente función logarítmica:

$$I(x_k) = \log\left(\frac{1}{p_k}\right) = -\log p_k \quad (\text{A2.4})$$



donde la base del logaritmo es arbitraria. Cuando se usa el logaritmo neperiano, la unidad de información es el *nat*, y cuando se usa el logaritmo en base 2 la unidad es el *bit*. En cualquier caso, la definición de información dada en (A2.4) cumple las siguientes propiedades:

PA2.1 Obviamente, si hay total certeza de que ocurra un evento, no se gana información con su producción:

$$I(x_k) = 0 \quad \text{para} \quad p_k = 1 \quad (\text{A2.5})$$

PA2.2 La generación del evento $X=x_k$ proporciona, o no, información pero nunca hay pérdida:

$$I(x_k) \geq 0 \quad \text{para} \quad 0 \leq p_k \leq 1 \quad (\text{A2.6})$$

PA2.3 Cuanto menos probable sea un evento más información sobre él se gana:

$$I(x_k) > I(x_i) \quad \text{para} \quad p_k < p_i \quad (\text{A2.7})$$

La cantidad de información $I(x_k)$ es una variable aleatoria discreta con probabilidad p_k . El valor medio de $I(x_k)$ en el rango de $2k + 1$ valores discretos viene dado por:

$$H(X) = E\{I(x_k)\} = \sum_{k=-K}^K p_k I(x_k) = - \sum_{k=-K}^K p_k \log p_k \quad (\text{A2.8})$$

La cantidad $H(X)$ se denomina *entropía* de la variable aleatoria X , y puede tomar un conjunto finito de valores discretos. Este nombre se debe a la analogía entre la definición dada en (A2.8) y la de la entropía dada en termodinámica como medida de desorden. La entropía $H(X)$ es una medida de la *media de la cantidad de información que lleva un mensaje*. Nótese, sin embargo, que la X en $H(X)$ no es un argumento de la función, sino más bien una etiqueta para asociarla con una variable aleatoria. Obsérvese también que en la definición dada en (A2.8) se toma $0 \log 0$ como 0.

La entropía $H(X)$ viene acotada por:

$$0 \leq H(X) \leq \log(2k + 1) \quad (\text{A2.9})$$

donde $(2k + 1)$ es el número total de niveles discretos. Además, se cumple lo siguiente:

PA2.4 $H(X)=0$ si y sólo si la probabilidad $p_k=1$ para algún k , y el resto de las probabilidades son cero. Esta cota inferior corresponde en entropía a la *no-incertidumbre* (seguridad en la producción de un evento).

PA2.5 $H(X)=\log_2(2k + 1)$, si y sólo si $p_k=1/(2k + 1)$ para algún k (es decir, todos los niveles discretos son equiprobables). Esta cota superior corresponde en entropía a un *máximo de incertidumbre*. La producción de ningún evento está privilegiada.

Sean $p_X(x)$ y $q_X(x)$ las probabilidades de que la variable aleatoria X pase al estado x bajo dos condiciones distintas. La *entropía relativa o divergencia Kullback-Leibler* (distancia)

entre dos *funciones de densidad de probabilidad* $p_X(x)$ y $q_X(x)$ se define como [KUL68], [COV91]:

$$D_{p||q} = \sum_{x \in \mathcal{X}} p_X(x) \log \left(\frac{p_X(x)}{q_X(x)} \right) \quad (\text{A2.10})$$

donde la sumatoria se extiende sobre todos los posibles estados del sistema. La función de densidad de probabilidad $q_X(x)$ desempeña el papel de *medida de referencia*. La divergencia Kullback-Leibler es una medida de la desigualdad entre dos distribuciones de probabilidad (siendo 0 cuando las distribuciones son iguales).

A2.2 ENTROPÍA DIFERENCIAL Y VARIABLES ALEATORIAS CONTINUAS.

Los conceptos sobre teoría de la información vistos implican a conjuntos de variables aleatorias cuyas amplitudes toman valores discretos. A continuación se amplían algunos de estos conceptos a variables aleatorias continuas.

Sea la variable aleatoria X con *función de densidad de probabilidad* $f_X(x)$. Por analogía con la entropía de una variable aleatoria discreta, se puede dar la siguiente definición:

$$h(X) = - \int_{-\infty}^{\infty} f_X(x) \log f_X(x) dx = -E\{\log f_X(x)\} \quad (\text{A2.11})$$

Se denomina a $h(X)$ *entropía diferencial* de X para distinguirla de la *entropía absoluta*. Aunque $h(X)$ es una cantidad matemática útil, no es en ningún sentido una medida de la aleatoriedad de X .

Si se tiene un vector aleatorio continuo \mathbf{X} formado por n variables aleatorias X_1, X_2, \dots, X_n , se define la *entropía diferencial conjunta* de \mathbf{X} como:

$$h(\mathbf{X}) = - \int_{-\infty}^{\infty} f_{\mathbf{X}}(\mathbf{x}) \log f_{\mathbf{X}}(\mathbf{x}) d\mathbf{x} = -E\{\log f_{\mathbf{X}}(\mathbf{x})\} \quad (\text{A2.12})$$

donde $f_{\mathbf{X}}(\mathbf{x})$ es la función de densidad de probabilidad conjunta de \mathbf{X} .

Propiedades de la entropía diferencial

PA2.6 De la definición de entropía diferencial, $h(X)$, dada en (A2.22) se concluye que una traslación, c , no cambia el valor:

$$h(X + c) = h(X) \quad (\text{A2.13})$$

donde c es una constante.

PA2.7 Otra propiedad útil es:

$$h(aX) = h(X) + \log|a| \quad (\text{A2.14})$$

donde a es un factor de escala.

PA2.8 La expresión (A2.14) se aplica a una variable aleatoria escalar, pero se puede generalizar al caso de un vector aleatorio \mathbf{X} multiplicado por una matriz \mathbf{A} :

$$h(\mathbf{AX}) = h(\mathbf{X}) + \log|\det(\mathbf{A})| \quad (\text{A2.15})$$

donde $\det(\mathbf{A})$ es el determinante de la matriz \mathbf{A} .

PA2.9 La entropía de una variable aleatoria X uniformemente distribuida en un intervalo $[a, b]$ es:

$$h(X) = \log|a - b| \quad (\text{A2.16})$$

A2.3 PRINCIPIO DE MÁXIMA ENTROPÍA

Sea un sistema estocástico con un conjunto de estados conocidos pero con probabilidades desconocidas, y del que de algún modo se conocen ciertas *condiciones* de la distribución de probabilidades de sus estados. Estas condiciones pueden ser una serie de valores medios o cotas de esos valores. El problema es elegir un modelo de probabilidad óptimo en algún

sentido, dado este conocimiento a priori. Se puede encontrar un número infinito de posibles modelos que satisfagan estas condiciones.

El *principio de máxima entropía* resuelve este problema y se define de la siguiente forma [JAY57, 82]:

Cuando se efectúa una inferencia estadística basada en información incompleta, ésta debe ser obtenida a partir de la distribución de probabilidad que maximice la entropía, bajo las restricciones de la distribución.

En efecto, la noción de entropía define un tipo de medida en el espacio de distribución de probabilidades de forma que las distribuciones de mayor entropía son favorecidas respecto a las demás.

De lo anterior se deduce que el problema de maximización de la entropía se reduce a un problema de optimización. Para ilustrar este procedimiento, se considera la maximización de la entropía diferencial:

$$h(X) = - \int_{-\infty}^{\infty} f_X(x) \log f_X(x) dx$$

respecto a todas las funciones de densidad de probabilidad $f_X(x)$ de una variable aleatoria X , bajo las siguientes condiciones:

1. $f_X(x) \geq 0$, dándose la igualdad fuera del rango de x .
2. $\int_{-\infty}^{\infty} f_X(x) dx = 1$
3. $\int_{-\infty}^{\infty} f_X(x) g_i(x) dx = \alpha_i$ para $i=1,2,\dots,m$

donde $g_i(x)$ es una función de x . Las condiciones 1 y 2 describen dos propiedades fundamentales de una función de densidad de probabilidad. La condición 3 tiene en cuenta el conocimiento a priori sobre la variable x .

Supóngase que el conocimiento a priori viene dado por la media, μ , y la varianza, σ^2 , de la variable aleatoria X . Por definición:

$$\int_{-\infty}^{\infty} (x - \mu)^2 f_X(x) dx = \sigma^2 = \text{constante} \quad (\text{A2.17})$$

Se puede demostrar que:

PA2.10 Para una varianza dada, σ^2 , la variable aleatoria gaussiana tiene la mayor entropía diferencial posible para cualquier variable aleatoria. Es decir, si X es una variable aleatoria gaussiana e Y es cualquier otra variable aleatoria con la misma media y varianza, entonces para todo Y se verifica:

$$h(X) \geq h(Y)$$

dándose la igualdad solamente si X e Y son iguales.

PA2.11 La entropía de una variable aleatoria gaussiana X viene determinada solamente por la varianza de X (es decir, es independiente de la media de X).

A2.4 INFORMACIÓN MUTUA

En el diseño de un sistema auto-organizativo, el principal objetivo es desarrollar un algoritmo que pueda aprender una relación entrada/salida de interés sólo a partir de los patrones de entrada. En este contexto la noción de información mutua es de gran importancia, debido a una serie de propiedades altamente deseables. Sea un sistema estocástico con entrada X y salida Y (Figura A2.1). Ambos sólo pueden tomar valores discretos, x e y respectivamente. La entropía $H(X)$ es una medida de la incertidumbre a priori de X . Para medir la incertidumbre de X dado Y se define la *entropía condicional de X dado Y* [GRA90], [COV91]:

$$H(X|Y) = H(X, Y) - H(Y) \quad (\text{A2.18})$$

con:

$$0 \leq H(X|Y) \leq H(X) \quad (\text{A2.19})$$

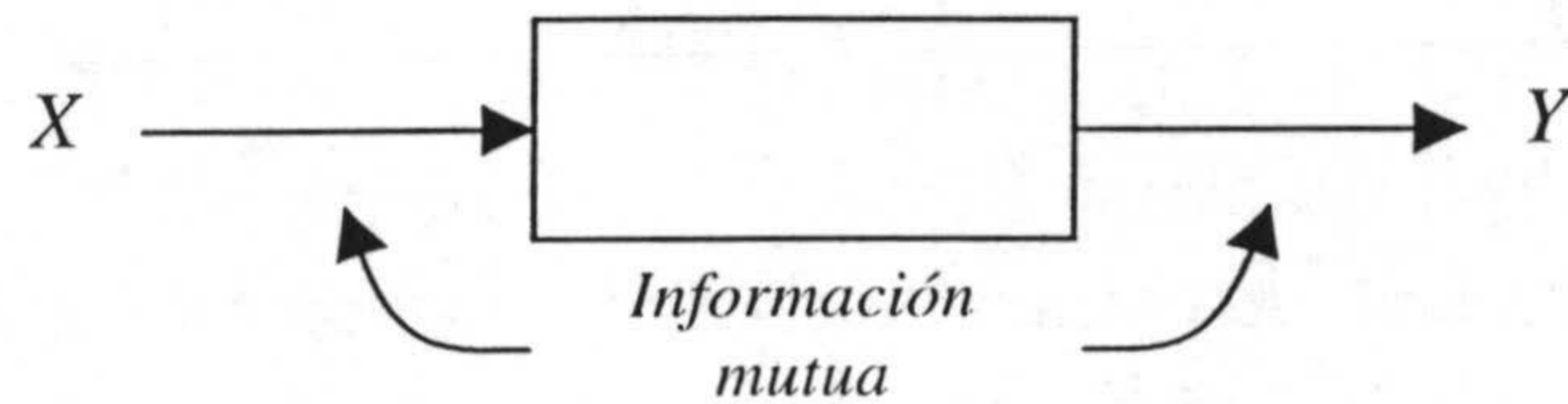


Figura A2.1 Información mutua.

La *entropía condicional* $H(X|Y)$ representa la cantidad de incertidumbre que queda sobre la entrada del sistema X después de haber observado la salida Y . La otra cantidad, $H(X,Y)$ en (A2.18), es la *entropía conjunta* de X e Y , que se define como:

$$H(X,Y) = - \sum_{x \in \mathcal{X}} \sum_{y \in \mathcal{Y}} p(x,y) \log \left(\frac{p(x,y)}{p(x)p(y)} \right) \quad (\text{A2.20})$$

donde $p(x,y)$ es la función de probabilidad conjunta de las variables discretas X e Y , y \mathcal{X} e \mathcal{Y} sus respectivos alfabetos (conjunto de valores posibles de X e Y , respectivamente). La entropía conjunta es la resultante de considerar simultáneamente X e Y .

Como la entropía $H(X)$ representa la incertidumbre de la entrada del sistema antes de observar la salida del sistema, y la entropía condicional $H(X|Y)$ representa la incertidumbre del sistema después de observar la salida del sistema, la diferencia $H(X) - H(X|Y)$ debe representar la *incertidumbre de la entrada que se va a obtener observando la salida*. Esta cantidad se denomina *información mutua* entre las variables aleatorias X e Y . Denotándola como $I(X;Y)$ se tiene:

$$I(X;Y) = H(X) - H(X|Y) = \sum_{x \in \mathcal{X}} \sum_{y \in \mathcal{Y}} p(x,y) \log \left(\frac{p(x,y)}{p(x)p(y)} \right) \quad (\text{A2.21})$$

La entropía es un caso especial de información mutua, ya que:

$$H(X) = I(X;X)$$

La información mutua $I(X;Y)$ entre dos variables aleatorias discretas, X e Y , tiene las siguientes propiedades [COV91], [GRA90]:

PA2.11 La información mutua entre X e Y es simétrica:

$$I(Y;X) = I(X;Y)$$

donde la información mutua $I(Y;X)$ es una medida de la incertidumbre de la salida Y obtenida observando la entrada X , y la información mutua $I(X;Y)$ es una medida de la incertidumbre de la entrada X obtenida observando la salida Y .

PA2.12 La información mutua entre X e Y es siempre no-negativa, es decir:

$$I(X;Y) \geq 0$$

En efecto, esta propiedad establece que no se puede perder información, en media, observando la salida del sistema Y . Además, la información mutua es 0 si y sólo si la entrada y la salida son estadísticamente independientes.

PA2.13 La información mutua entre X e Y puede expresarse en términos de la entropía de Y :

$$I(X;Y) = H(Y) - H(Y|X) \quad (\text{A2.22})$$

donde $H(Y|X)$ es la entropía condicional. El segundo miembro de la (A2.22) es la media de la información dada por la salida del sistema Y , menos la media de la información dada por Y suponiendo que conocemos la entrada X . La última cantidad, $H(Y|X)$, da más bien información sobre el ruido del proceso, que sobre la entrada X .

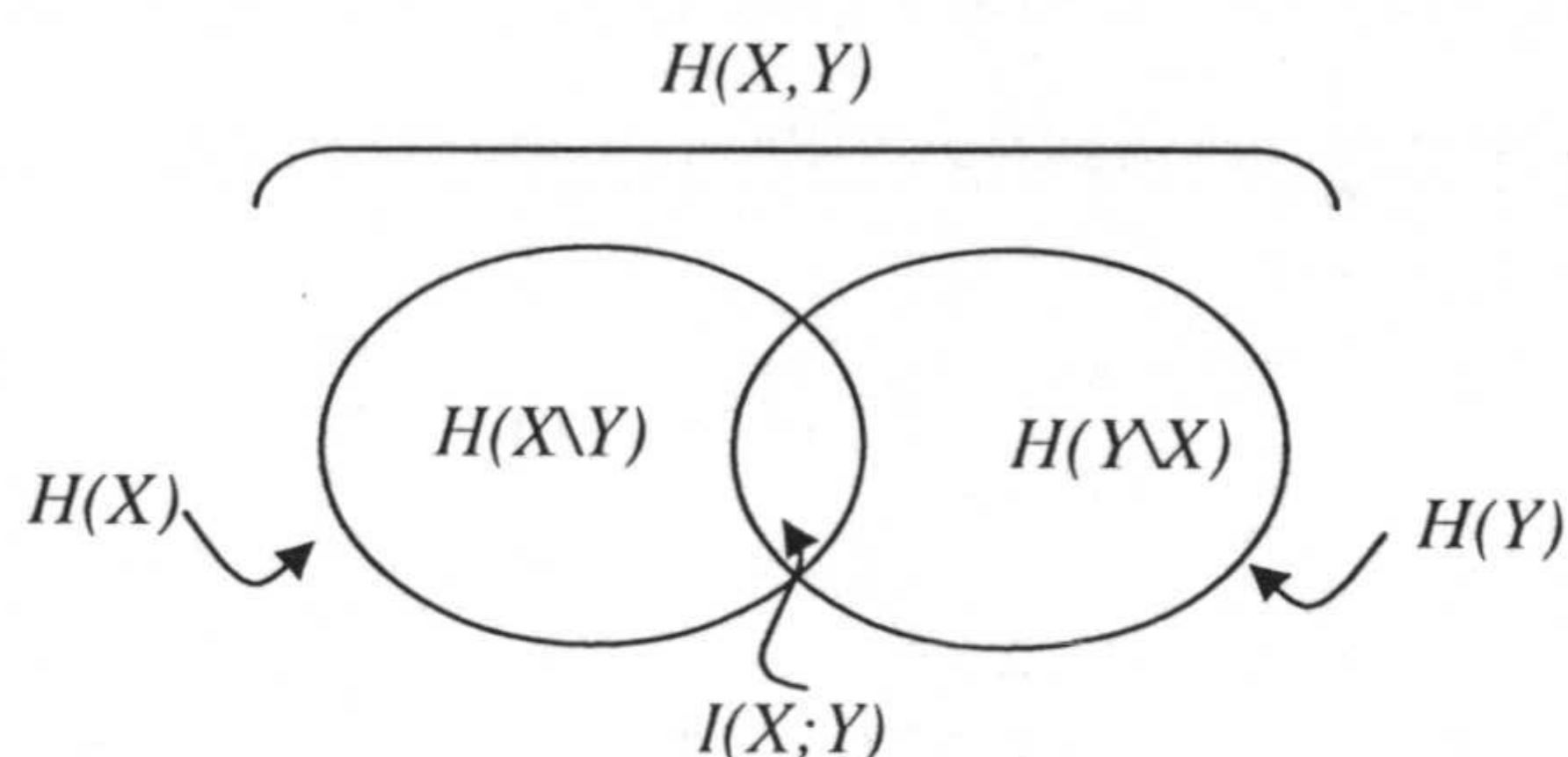


Figura A2.2 Relación entre la información mutua $I(X;Y)$ y las entropías $H(X)$ y $H(Y)$.

La Figura A2.2 ofrece una interpretación visual de (A2.21) y (A2.22). La entropía de la entrada X se representa con el círculo de la izquierda. La entropía de la salida Y se representa con el círculo de la derecha. La información mutua entre X e Y se representa con la intersección de ambos círculos.

A2.5 INFORMACIÓN MUTUA PARA VARIABLES ALEATORIAS CONTINUAS.

Sean un par de variables aleatorias continuas X e Y . Por analogía con (A2.21) se define la información mutua entre las variables aleatorias X e Y como:

$$I(X;Y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f_{X,Y}(x,y) \log \left(\frac{f_X(x|y)}{f_X(x)} \right) dx dy \quad (\text{A2.23})$$

donde $f_{X,Y}(x,y)$ es la función de densidad de probabilidad conjunta de X e Y , y $f_{X,Y}(x|y)$ es la función de densidad de probabilidad condicional de X , dado $Y = y$. Nótese que (A1.18):

$$f_{X,Y}(x,y) = f_{X,Y}(x|y) f_Y(y)$$

y por tanto:

$$I(X;Y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f_{X,Y}(x,y) \log \left(\frac{f_{X,Y}(x,y)}{f_X(x) f_Y(y)} \right) dx dy \quad (\text{A2.24})$$

También, por analogía con las variables aleatorias discretas, la información mutua $I(X;Y)$ entre dos variables aleatorias continuas, X e Y tiene las siguientes propiedades:

$$I(X;Y) = h(X) - h(X|Y) = h(Y) - h(Y|X) = h(X) + h(Y) - h(X,Y) \quad (\text{A2.25})$$

$$I(Y;X) = I(X;Y) \quad (\text{A2.26})$$

$$I(X;Y) \geq 0 \quad (\text{A2.27})$$

El parámetro $h(X)$ es la entropía diferencial de X ; lo mismo para $h(Y)$. El parámetro $h(X|Y)$ es la entropía diferencial condicional de X dado Y , que se define:

$$h(X|Y) = - \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f_{X,Y}(x,y) \log f_X(x|y) dx dy \quad (\text{A2.28})$$

El parámetro $h(Y|X)$ es la entropía diferencial condicional de Y dado X , y se define de forma similar a $h(X|Y)$. El parámetro $h(X,Y)$ es la entropía diferencial conjunta de X e Y .

Nótese que en (A2.27) se da la igualdad sólo cuando las variables aleatorias X e Y son estadísticamente independientes. Cuando se satisface esta condición, la densidad de probabilidad conjunta de X e Y se puede factorizar (A1.21):

$$f_{X,Y}(x,y) = f_X(x) f_Y(y) \quad (\text{A2.29})$$

donde $f_X(x)$ y $f_Y(y)$ son las funciones de densidad de probabilidad marginales de X e Y . De la misma forma se puede escribir:

$$f_{X,Y}(x|y) = f_X(x) \quad (\text{A2.30})$$

lo que quiere decir que el conocimiento de Y no puede afectar de ninguna forma a la distribución de X . Aplicando esta condición a (A2.23), la información mutua $I(X;Y)$ entre X e Y se reduce a 0.

La definición de información mutua $I(X;Y)$ dada en (A2.23) se aplica a variables aleatorias escalares X e Y . Esta definición se puede ampliar a vectores aleatorios \mathbf{X} e \mathbf{Y} , y se puede definir $I(\mathbf{X};\mathbf{Y})$ como:

$$I(\mathbf{X};\mathbf{Y}) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f_{\mathbf{X},\mathbf{Y}}(\mathbf{x},\mathbf{y}) \log \left(\frac{f_{\mathbf{X}}(\mathbf{x}|\mathbf{y})}{f_{\mathbf{X}}(\mathbf{x})} \right) d\mathbf{x}d\mathbf{y} \quad (\text{A2.31})$$

La información mutua $I(\mathbf{X};\mathbf{Y})$ tiene propiedades equivalentes a las dadas en (A2.25) a (A2.27), para variables aleatorias escalares.

A2.6 DIVERGENCIA KULLBACK-LEIBLER

En (A2.10) se definió la divergencia Kullback-Leibler para variables aleatorias discretas. Esta definición se puede ampliar al caso general de vectores aleatorios continuos. Sean $f_X(\mathbf{x})$ y $g_X(\mathbf{x})$ dos funciones con densidades de probabilidad distintas, de un vector aleatorio

($m \times 1$), \mathbf{X} . Como en (A2.10) se puede definir la divergencia Kullback-Leibler entre $f_{\mathbf{X}}(\mathbf{x})$ y $g_{\mathbf{X}}(\mathbf{x})$ ([KUL68], [SHO80]):

$$D_{f_{\mathbf{X}} \parallel g_{\mathbf{X}}} = \int_{-\infty}^{\infty} f_{\mathbf{X}}(\mathbf{x}) \log \left(\frac{f_{\mathbf{X}}(\mathbf{x})}{g_{\mathbf{X}}(\mathbf{x})} \right) d\mathbf{x} \quad (\text{A2.32})$$

La divergencia Kullback-Leibler tiene las siguientes propiedades:

PA1.14 Su valor es siempre positivo o cero. Para el caso especial en el que $f_{\mathbf{X}}(\mathbf{x}) = g_{\mathbf{X}}(\mathbf{x})$ las dos distribuciones son idénticas y $D_{f \parallel g}$ es cero.

PA1.15 Permanece invariable ante los siguientes cambios en las componentes del vector \mathbf{x} :

- Permutaciones en el orden de las componentes.
- Cambios de escala en la amplitud.
- Transformaciones no lineales monótonas.

La información mutua $I(\mathbf{X}; \mathbf{Y})$ entre un par de vectores, \mathbf{X} e \mathbf{Y} tiene una interpretación interesante en términos de la divergencia Kullback-Leibler. Primero, nótese que:

$$f_{\mathbf{X}, \mathbf{Y}}(\mathbf{x}, \mathbf{y}) = f_{\mathbf{Y}}(\mathbf{y} | \mathbf{x}) f_{\mathbf{X}}(\mathbf{x}) \quad (\text{A2.33})$$

Por tanto, la forma equivalente a (A2.31) es:

$$I(\mathbf{X}; \mathbf{Y}) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f_{\mathbf{X}, \mathbf{Y}}(\mathbf{x}, \mathbf{y}) \log \left(\frac{f_{\mathbf{X}, \mathbf{Y}}(\mathbf{x}, \mathbf{y})}{f_{\mathbf{X}}(\mathbf{x}) f_{\mathbf{Y}}(\mathbf{y})} \right) d\mathbf{x} d\mathbf{y} \quad (\text{A2.34})$$

Comparando (A2.34) con (A2.32) se puede deducir:

$$I(\mathbf{X}; \mathbf{Y}) = D_{f_{\mathbf{X}, \mathbf{Y}} \parallel f_{\mathbf{X}} f_{\mathbf{Y}}} \quad (\text{A2.35})$$

Es decir, la información mutua $I(\mathbf{X}; \mathbf{Y})$ entre \mathbf{X} e \mathbf{Y} es igual a la divergencia Kullback-Leibler entre la función de densidad de probabilidad $f_{\mathbf{X}}(\mathbf{x})$ de un vector aleatorio ($m \times 1$), \mathbf{X} ,

y el producto de sus m funciones de densidad de probabilidad marginales. Sea $\tilde{f}_{x_i}(x_i)$ la i -ésima función de densidad de probabilidad marginal del elemento X_i , que se define:

$$\tilde{f}_{x_i}(x_i) = \int_{-\infty}^{\infty} f_{\mathbf{X}}(\mathbf{x}) dx^{(i)}, \quad i=1,2,\dots,m \quad (\text{A2.36})$$

donde $\mathbf{x}^{(i)}$ es el vector $(m-1) \times 1$ que queda después de eliminar el i -ésimo elemento del vector \mathbf{x} .

La divergencia Kullback-Leibler entre $f_{\mathbf{X}}(\mathbf{x})$ y la distribución factorial $\prod_{i=1}^m \tilde{f}_{x_i}(x_i)$ viene dada por:

$$D_{f_{\mathbf{X}} \parallel \tilde{f}_{\mathbf{X}}} = \int_{-\infty}^{\infty} f_{\mathbf{X}}(\mathbf{x}) \log \left(\frac{f_{\mathbf{X}}(\mathbf{x})}{\prod_{i=1}^m \tilde{f}_{x_i}(x_i)} \right) dx \quad (\text{A2.37})$$

que también se puede expresar en forma de desarrollo en serie:

$$D_{f_{\mathbf{X}} \parallel \tilde{f}_{\mathbf{X}}} = \int_{-\infty}^{\infty} f_{\mathbf{X}}(\mathbf{x}) \log f_{\mathbf{X}}(\mathbf{x}) dx - \sum_{i=1}^m \int_{-\infty}^{\infty} f_{\mathbf{X}}(\mathbf{x}) \log \tilde{f}_{x_i}(x_i) dx \quad (\text{A2.38})$$

La expresión (A2.38) se puede simplificar resultando ser:

$$D_{f_{\mathbf{X}} \parallel \tilde{f}_{\mathbf{X}}} = -h(\mathbf{X}) + \sum_{i=1}^m \tilde{h}(X_i) \quad (\text{A2.39})$$

Esta fórmula es interesante en el estudio del problema de separación ciega de fuentes.

A2.7 DESCOMPOSICIÓN PITAGÓREA

Considérese la divergencia Kullback-Leibler entre las funciones de densidad de probabilidad $f_{\mathbf{X}}(\mathbf{x})$ y $f_{\mathbf{U}}(\mathbf{x})$. El vector aleatorio $(m \times 1)$ \mathbf{U} está formado por variables independientes, de forma que:

$$f_{\mathbf{U}}(\mathbf{x}) = \prod_{i=1}^m f_{U_i}(x_i)$$

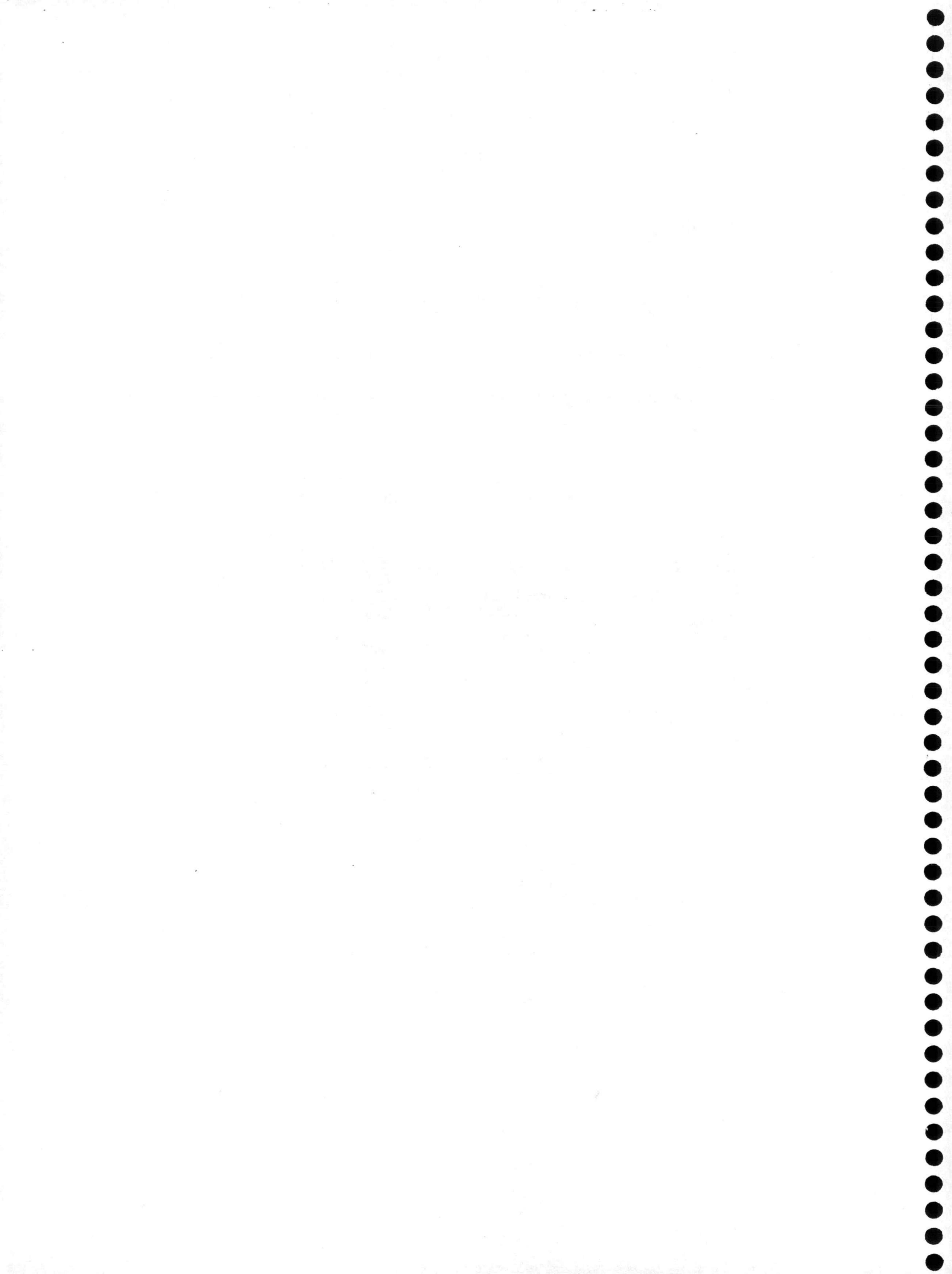
y el vector aleatorio ($m \times 1$) \mathbf{X} se define en función de \mathbf{U} :

$$\mathbf{X} = \mathbf{A}\mathbf{U}$$

donde \mathbf{A} es una matriz no diagonal. Sea $\tilde{f}_{x_i}(x_i)$ la función de densidad de probabilidad marginal de cada X_i obtenido de $f_{\mathbf{X}}(\mathbf{x})$. Entonces, la divergencia Kullback-Leibler entre $f_{\mathbf{X}}(\mathbf{x})$ y $f_{\mathbf{U}}(\mathbf{x})$ admite la siguiente descomposición Pitagórea:

$$D_{f_{\mathbf{X}} \parallel f_{\mathbf{U}}} = D_{f_{\mathbf{X}} \parallel \tilde{f}_{\mathbf{X}}} + D_{\tilde{f}_{\mathbf{X}} \parallel f_{\mathbf{U}}} \quad (\text{A2.40})$$

Esta descomposición Pitagórea tiene una interpretación geométrica análoga al teorema clásico de Pitágoras, de ahí su nombre ([AMA85]).



APÉNDICE 3:

RESULTADOS DE SIMULACIÓN OBTENIDOS CON EL ALGORITMO DE LOCALIZACIÓN DE LOS EJES DE LA DISTRIBUCIÓN DE VECTORES DE MEZCLA.

En este apéndice se incluyen los resultados de simulación obtenidos con el algoritmo de localización de los ejes de distribución de vectores de mezcla, descrito en el Capítulo 5.

En todos los casos las fuentes han sido señales estandarizadas con funciones de distribución de densidad de probabilidad gamma bilateral, y el ruido de tipo uniforme.

Las referencias a tablas y figuras tienen el siguiente formato: Tabla A3.M.N o Figura A3.M.N, donde A3 significa Apéndice 3, M (1,2,...,8) identifica la matriz de mezcla (A_1 a A_8) referida en cada tabla o figura, y N es el número de orden correlativo de la tabla o figura.

El significado y discusión de estas tablas y figuras se encuentra en la Sec.5.6 del Capítulo 5.

A3.1 RESULTADOS OBTENIDOS CON LA MATRIZ A_1

Tabla A3.1.1. Valor medio del error de estimación de la matriz A_1 (para cada caso se consideran 10 medidas).

m	% Ruido						
	0	1	2	3	4	5	6
10.000	0.0003	0.0040	0.0130	0.0202	0.0351	0.0503	0.0345
20.000	0.0004	0.0045	0.0145	0.0152	0.0351	0.0320	0.0217
30.000	0.0005	0.0035	0.0118	0.0147	0.0231	0.0376	0.0193
40.000	0.0005	0.0040	0.0097	0.0178	0.0299	0.0450	0.0373
50.000	0.0005	0.0051	0.0102	0.0163	0.0210	0.0357	0.0176
60.000	0.0005	0.0040	0.0087	0.0179	0.0215	0.0386	0.0252
70.000	0.0005	0.0021	0.0086	0.0142	0.0221	0.0376	0.0252
80.000	0.0005	0.0031	0.0091	0.0178	0.0254	0.0347	0.0315
90.000	0.0005	0.0016	0.0081	0.0163	0.0253	0.0302	0.0155
100.000	0.0005	0.0016	0.0101	0.0184	0.0248	0.0365	0.0179
110.000	0.0005	0.0016	0.0107	0.0184	0.0241	0.0296	0.0129
120.000	0.0005	0.0016	0.0106	0.0205	0.0264	0.0385	0.0221
130.000	0.0005	0.0012	0.0091	0.0162	0.0252	0.0296	0.0182
140.000	0.0005	0.0016	0.0086	0.0167	0.0241	0.0364	0.0179
150.000	0.0005	0.0012	0.0075	0.0157	0.0258	0.0290	0.0194
160.000	0.0005	0.0012	0.0076	0.0161	0.0263	0.0364	0.0182
170.000	0.0005	0.0016	0.0071	0.0162	0.0262	0.0307	0.0219
180.000	0.0005	0.0012	0.0080	0.0166	0.0258	0.0376	0.0231
190.000	0.0005	0.0012	0.0080	0.0150	0.0270	0.0319	0.0189
200.000	0.0005	0.0021	0.0085	0.0183	0.0308	0.0376	0.0212

Tabla A3.1.2. Desviaciones estándar de los valores medios del error.

m	% Ruido						
	0	1	2	3	4	5	6
10.000	0	0.0024	0.0069	0.0120	0.0288	0.0449	0.0606
20.000	0	0.0039	0.0063	0.0075	0.0234	0.0223	0.0359
30.000	0	0.0025	0.0022	0.0103	0.0157	0.0321	0.0332
40.000	0	0.0032	0.0052	0.0094	0.0163	0.0281	0.0732
50.000	0	0.0027	0.0074	0.0111	0.0131	0.0310	0.0293
60.000	0	0.0039	0.0065	0.0073	0.0144	0.0283	0.0468
70.000	0	0.0022	0.0056	0.0086	0.0143	0.0307	0.0468
80.000	0	0.0034	0.0055	0.0063	0.0112	0.0208	0.0615
90.000	0	0.0020	0.0061	0.0081	0.0148	0.0226	0.0273
100.000	0	0.0020	0.0052	0.0057	0.0144	0.0214	0.0335
110.000	0	0.0020	0.0056	0.0096	0.0136	0.0218	0.0212
120.000	0	0.0020	0.0050	0.0072	0.0122	0.0278	0.0404
130.000	0	0.0015	0.0056	0.0084	0.0146	0.0222	0.0332
140.000	0	0.0020	0.0057	0.0069	0.0135	0.0198	0.0335
150.000	0	0.0015	0.0036	0.0089	0.0142	0.0221	0.0315
160.000	0	0.0015	0.0043	0.0080	0.0112	0.0217	0.0307
170.000	0	0.0020	0.0042	0.0091	0.0103	0.0227	0.0364
180.000	0	0.0015	0.0036	0.0074	0.0148	0.0218	0.0379
190.000	0	0.0015	0.0036	0.0087	0.0168	0.0236	0.0316
200.000	0	0.0032	0.0036	0.0077	0.0141	0.0218	0.0345

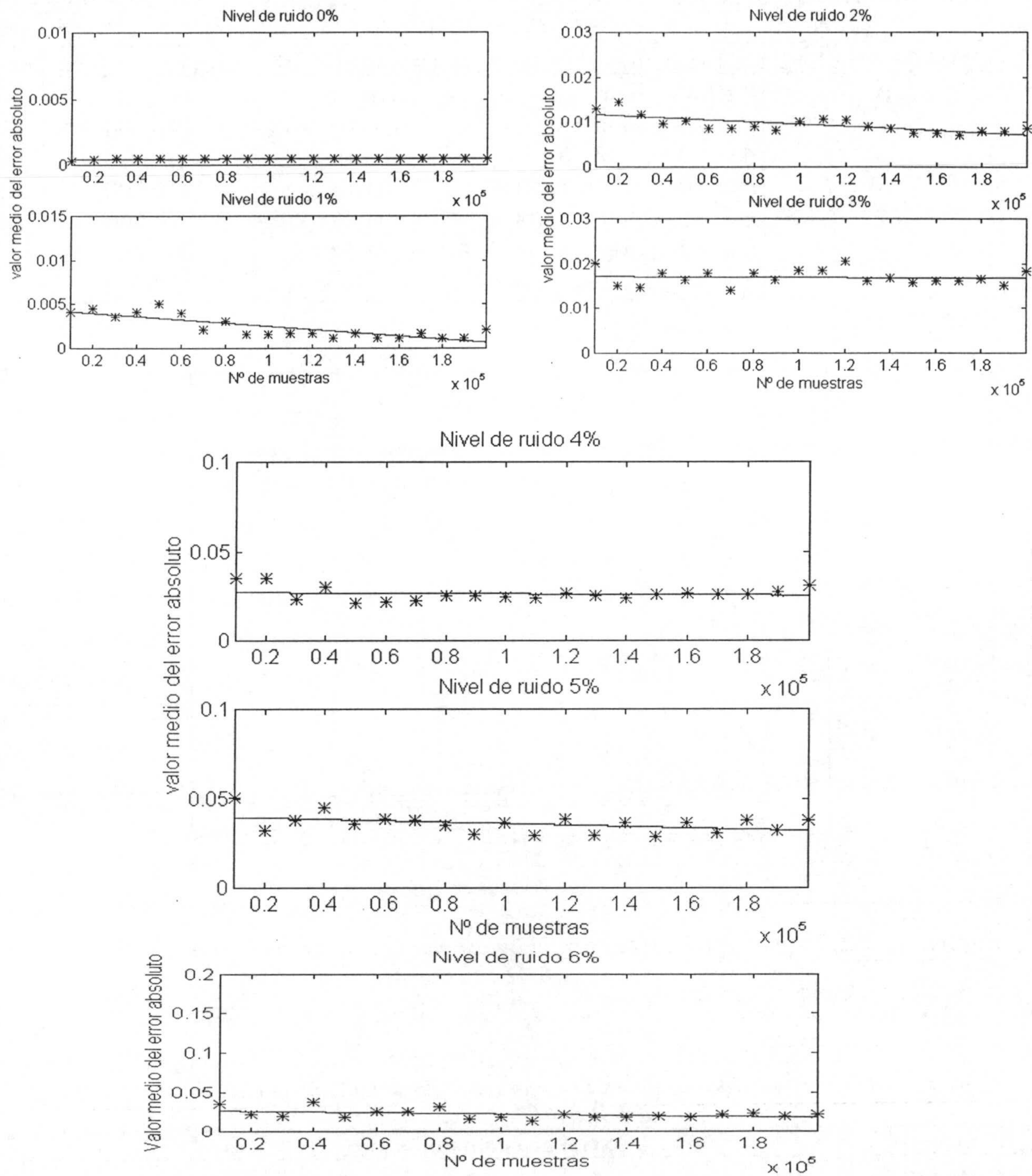


Figura A3.1.1. Valor medio del error absoluto frente al número de muestras para distintos niveles de ruido.

Tabla A3.1.3. Ecuaciones de las rectas de ajuste correspondientes a la Figura A3.1.1, valor del error absoluto para 100.000 muestras y n° de muestras para que el error absoluto sea 0.02.

Nivel de error (%)	Ecuación de la recta	ϵ para $m=100.000$	m_L para $\epsilon = 0.02$
0	$\epsilon = 0.04e-8 m + 0.0004$	0.00046	<10.000
1	$\epsilon = - 1.8e-8 m + 0.004$	0.0025	<10.000
2	$\epsilon = - 2.4e-8 m + 0.012$	0.0096	<10.000
3	$\epsilon = - 0.23e-8 m + 0.017$	0.017	<10.000
4	$\epsilon = - 0.9e-8 m + 0.03$	0.026	818.566
5	$\epsilon = - 3.9e-8 m + 0.04$	0.036	504.991
6	$\epsilon = - 4.5e-8 m + 0.03$	0.022	149.409

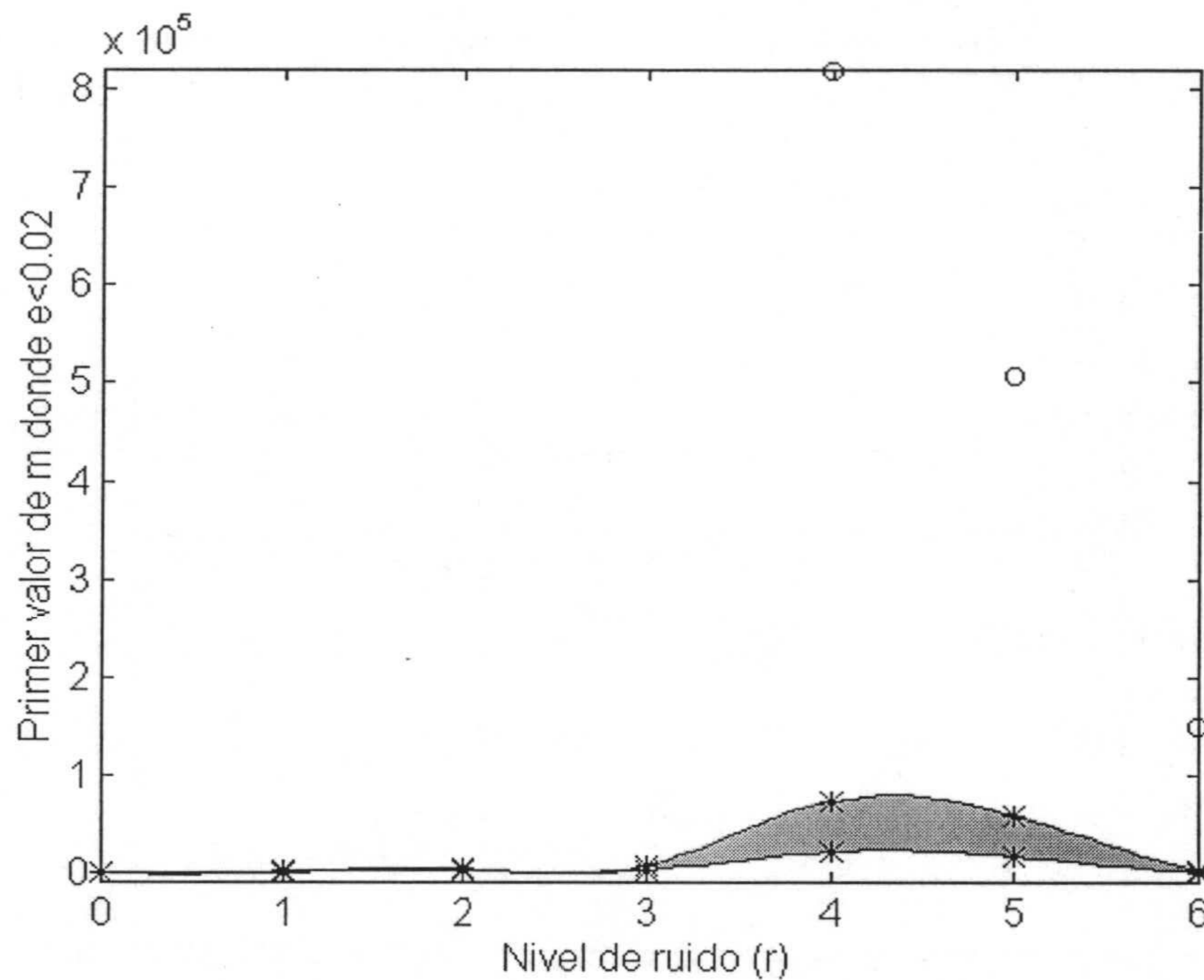


Figura A3.1.2. Primer valor de M para el que se obtiene un error absoluto menor de 0.02.

Tabla A3.1.4. Primer valor de m con un error absoluto de 0.02, con la matriz A_1 para los distintos niveles de ruido.

Matriz A_1	Nivel de ruido uniforme (%)						
	0	1	2	3	4	5	6
Valor medio	500	680	1760	2740	21560	16810	600
Desviación estándar	0	429	1661	4181	51263	42150	1192

A3.2 RESULTADOS OBTENIDOS CON LA MATRIZ A_2

Tabla A3.2.1. Valor medio del error de estimación de la matriz A_2 (para cada caso se consideran 10 medidas).

m	% Ruido						
	0	1	2	3	4	5	6
10.000	0.0024	0.0050	0.0091	0.0121	0.0223	0.0249	0.0355
20.000	0.0025	0.0040	0.0067	0.0101	0.0180	0.0237	0.0350
30.000	0.0024	0.0029	0.0066	0.0102	0.0170	0.0174	0.0244
40.000	0.0024	0.0030	0.0059	0.0109	0.0142	0.0252	0.0304
50.000	0.0024	0.0030	0.0064	0.0093	0.0156	0.0208	0.0280
60.000	0.0025	0.0029	0.0079	0.0082	0.0160	0.0238	0.0257
70.000	0.0025	0.0029	0.0076	0.0095	0.0152	0.0176	0.0280
80.000	0.0028	0.0029	0.0065	0.0097	0.0148	0.0174	0.0250
90.000	0.0028	0.0029	0.0066	0.0088	0.0169	0.0134	0.0220
100.000	0.0024	0.0029	0.0066	0.0087	0.0133	0.0202	0.0209
110.000	0.0023	0.0029	0.0062	0.0104	0.0137	0.0151	0.0230
120.000	0.0023	0.0029	0.0059	0.0091	0.0158	0.0190	0.0236
130.000	0.0023	0.0030	0.0069	0.0078	0.0137	0.0178	0.0199
140.000	0.0024	0.0029	0.0061	0.0082	0.0142	0.0232	0.0298
150.000	0.0023	0.0029	0.0066	0.0083	0.0168	0.0158	0.0205
160.000	0.0023	0.0029	0.0064	0.0068	0.0151	0.0232	0.0221
170.000	0.0023	0.0029	0.0064	0.0055	0.0142	0.0153	0.0198
180.000	0.0023	0.0029	0.0063	0.0052	0.0177	0.0233	0.0224
190.000	0.0023	0.0029	0.0063	0.0052	0.0142	0.0155	0.0208
200.000	0.0022	0.0029	0.0063	0.0066	0.0128	0.0218	0.0217

Tabla A3.2.2. Desviaciones estándar de los valores medios del error.

m	% Ruido						
	0	1	2	3	4	5	6
10.000	0	0.0035	0.0030	0.0059	0.0136	0.0095	0.0189
20.000	0	0.0017	0.0035	0.0037	0.0088	0.0127	0.0163
30.000	0	0.0002	0.0037	0.0057	0.0093	0.0165	0.0137
40.000	0	0.0002	0.0034	0.0071	0.0098	0.0118	0.0119
50.000	0	0.0002	0.0031	0.0024	0.0112	0.0192	0.0134
60.000	0	0.0001	0.0032	0.0035	0.0069	0.0119	0.0118
70.000	0	0.0002	0.0035	0.0036	0.0094	0.0161	0.0134
80.000	0	0.0002	0.0028	0.0033	0.0100	0.0108	0.0160
90.000	0	0.0001	0.0029	0.0052	0.0072	0.0118	0.0178
100.000	0	0.0001	0.0029	0.0047	0.0075	0.0099	0.0151
110.000	0	0.0001	0.0032	0.0036	0.0072	0.0122	0.0183
120.000	0	0.0002	0.0029	0.0043	0.0065	0.0082	0.0146
130.000	0	0.0002	0.0033	0.0031	0.0068	0.0154	0.0156
140.000	0	0.0001	0.0028	0.0050	0.0069	0.0100	0.0093
150.000	0	0.0001	0.0031	0.0041	0.0045	0.0133	0.0154
160.000	0	0.0001	0.0031	0.0044	0.0073	0.0108	0.0141
170.000	0	0.0001	0.0031	0.0022	0.0067	0.0135	0.0170
180.000	0	0.0001	0.0028	0.0024	0.0068	0.0095	0.0151
190.000	0	0.0001	0.0028	0.0024	0.0083	0.0133	0.0162
200.000	0	0.0001	0.0028	0.0047	0.0080	0.0098	0.0144

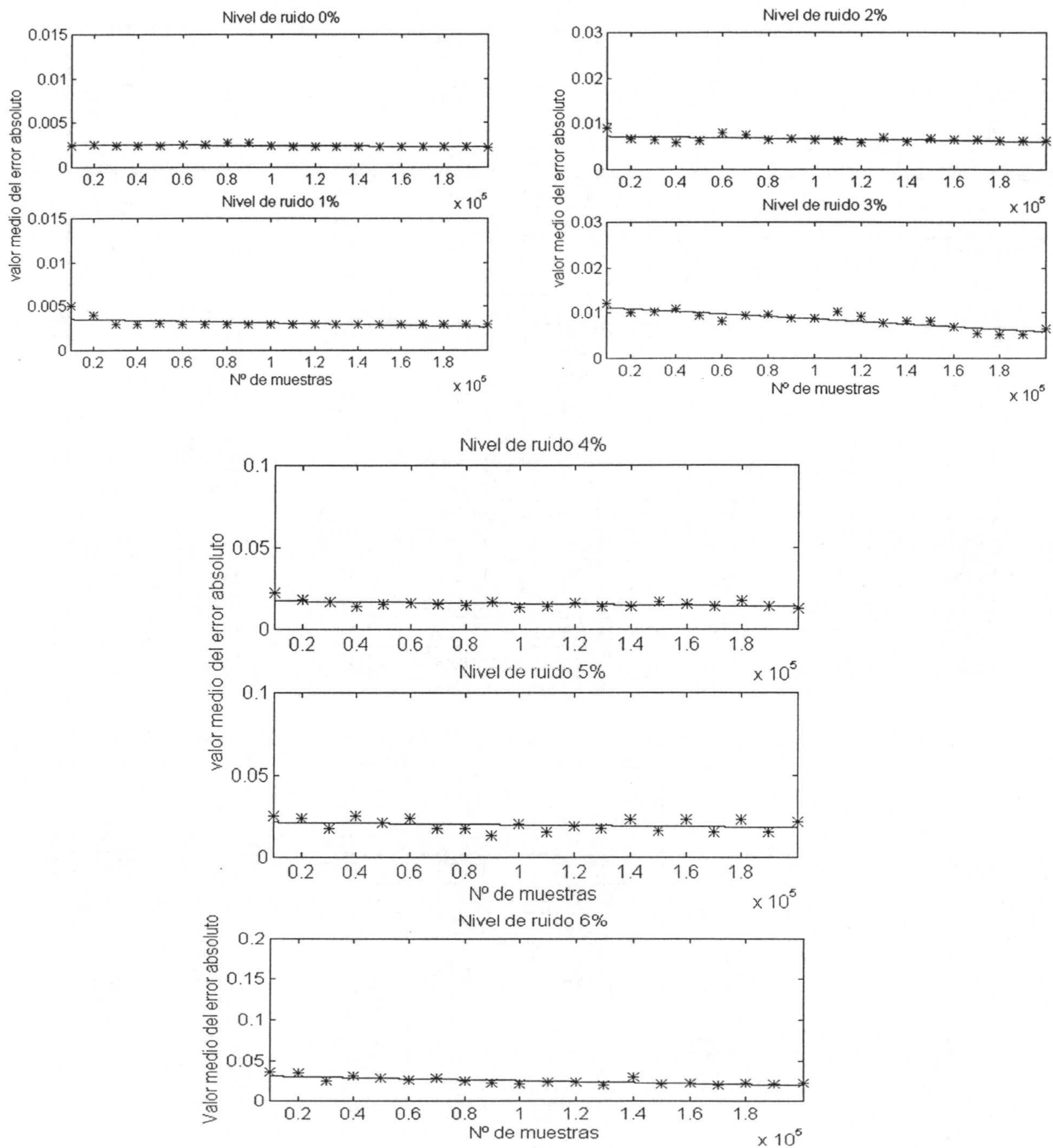


Figura A3.2.1. Valor medio del error absoluto frente al número de muestras para distintos niveles de ruido.

Tabla A3.2.3. Ecuaciones de las rectas de ajuste correspondientes a la Figura A3.2.1, valor del error absoluto para 100.000 muestras y n° de muestras para que el error absoluto sea 0.02.

Nivel de error (%)	Ecuación de la recta	ϵ para $m=100.000$	m_L para $\epsilon = 0.02$
0	$\epsilon = - 0.14e-8 m + 0.003$	0.0024	<10.000
1	$\epsilon = - 0.5e-8 m + 0.004$	0.0031	<10.000
2	$\epsilon = - 0.6e-8 m + 0.007$	0.0067	<10.000
3	$\epsilon = - 2.9e-8 m + 0.012$	0.0087	<10.000
4	$\epsilon = - 1.9e-8 m + 0.018$	0.0157	<10.000
5	$\epsilon = - 1.7e-8 m + 0.022$	0.0198	88971
6	$\epsilon = - 5.9e-8 m + 0.03$	0.0252	188331

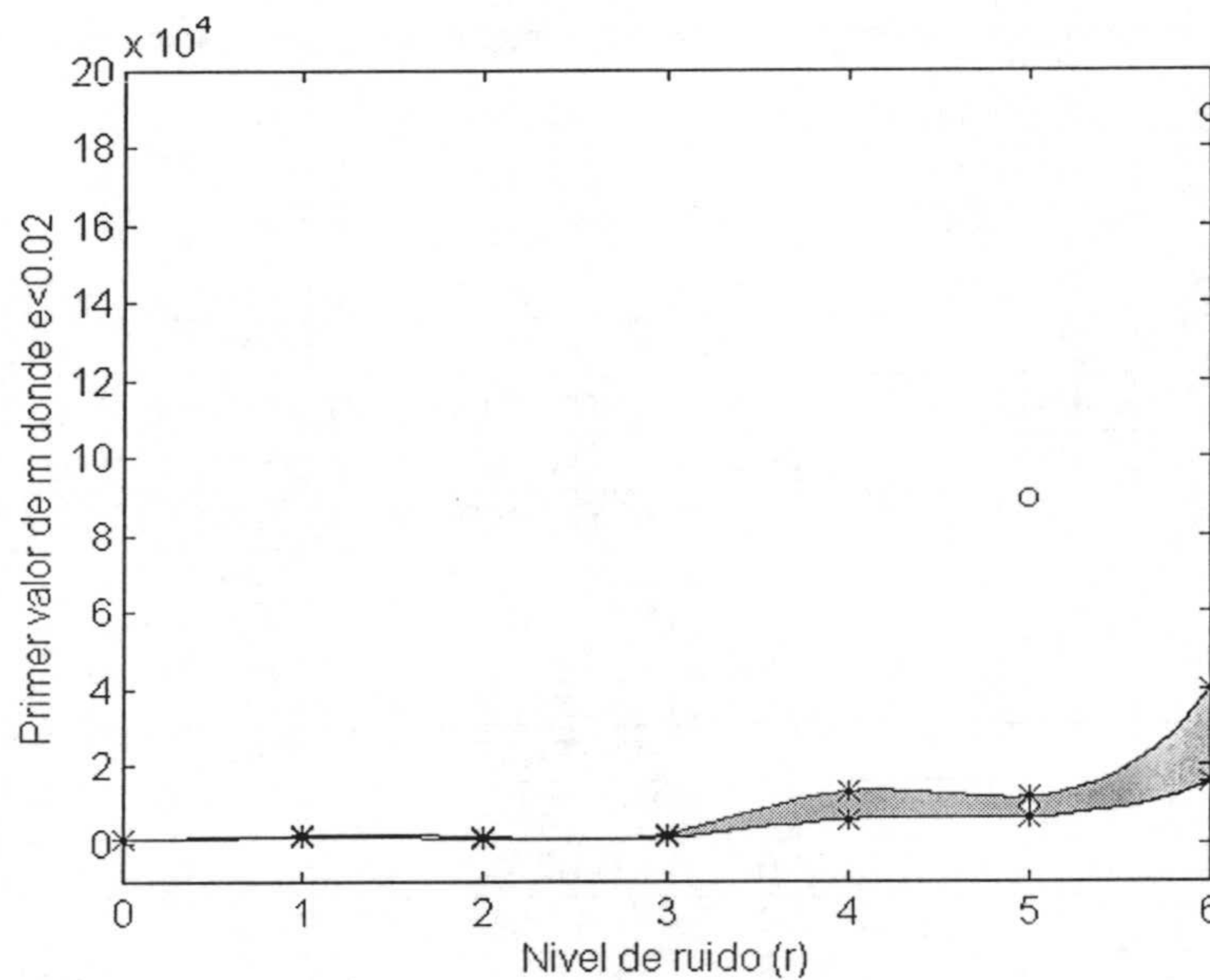


Figura A3.2.2. Primer valor de M donde el error absoluto es menor de 0.02.

Tabla A3.2.4. Primer valor de m con un error absoluto de 0.02, con la matriz A_2 para los distintos niveles de ruido.

	Nivel de ruido uniforme (%)						
	0	1	2	3	4	5	6
Matriz A_2	0	1	2	3	4	5	6
Valor medio	1000	1500	1200	1600	6200	6700	15800
Desviación estándar	0	527	422	966	7376	5229	24225

A3.3 RESULTADOS OBTENIDOS CON LA MATRIZ A_3

Tabla A3.3.1. Valor medio del error de estimación de la matriz A_3 (para cada caso se consideran 10 medidas).

m	% Ruido						
	0	1	2	3	4	5	6
10.000	0.0052	0.0055	0.0138	0.0170	0.0269	0.0282	0.0433
20.000	0.0051	0.0056	0.0094	0.0157	0.0205	0.0218	0.0375
30.000	0.0049	0.0056	0.0090	0.0137	0.0158	0.0274	0.0369
40.000	0.0050	0.0056	0.0082	0.0119	0.0184	0.0257	0.0374
50.000	0.0054	0.0056	0.0055	0.0109	0.0126	0.0240	0.0250
60.000	0.0052	0.0056	0.0084	0.0098	0.0138	0.0203	0.0236
70.000	0.0055	0.0056	0.0060	0.0130	0.0120	0.0128	0.0109
80.000	0.0055	0.0056	0.0056	0.0108	0.0136	0.0112	0.0185
90.000	0.0055	0.0056	0.0050	0.0095	0.0114	0.0100	0.0146
100.000	0.0056	0.0056	0.0061	0.0085	0.0124	0.0100	0.0180
110.000	0.0056	0.0056	0.0050	0.0092	0.0081	0.0109	0.0082
120.000	0.0056	0.0056	0.0056	0.0085	0.0102	0.0119	0.0263
130.000	0.0056	0.0056	0.0050	0.0097	0.0101	0.0108	0.0108
140.000	0.0056	0.0056	0.0055	0.0092	0.0118	0.0129	0.0197
150.000	0.0056	0.0055	0.0050	0.0092	0.0091	0.0138	0.0222
160.000	0.0055	0.0055	0.0055	0.0097	0.0114	0.0127	0.0193
170.000	0.0055	0.0055	0.0050	0.0097	0.0097	0.0138	0.0134
180.000	0.0055	0.0055	0.0055	0.0107	0.0113	0.0154	0.0182
190.000	0.0055	0.0055	0.0049	0.0102	0.0090	0.0137	0.0136
200.000	0.0054	0.0055	0.0055	0.0079	0.0108	0.0160	0.0201

Tabla A3.3.2. Desviaciones estándar de los valores medios del error.

m	% Ruido						
	0	1	2	3	4	5	6
10.000	0	0.0002	0.0066	0.0061	0.0103	0.0133	0.0160
20.000	0	0.0002	0.0046	0.0086	0.0096	0.0157	0.0155
30.000	0	0.0002	0.0048	0.0061	0.0117	0.0129	0.0307
40.000	0	0.0002	0.0028	0.0066	0.0104	0.0117	0.0241
50.000	0	0.0002	0.0026	0.0050	0.0086	0.0121	0.0302
60.000	0	0.0002	0.0057	0.0050	0.0057	0.0147	0.0192
70.000	0	0.0002	0.0031	0.0061	0.0084	0.0102	0.0158
80.000	0	0.0002	0.0002	0.0034	0.0068	0.0108	0.0186
90.000	0	0.0002	0.0018	0.0028	0.0081	0.0106	0.0174
100.000	0	0.0002	0.0018	0.0040	0.0075	0.0092	0.0189
110.000	0	0.0002	0.0018	0.0051	0.0069	0.0118	0.0114
120.000	0	0.0002	0.0002	0.0048	0.0066	0.0102	0.0161
130.000	0	0.0002	0.0018	0.0047	0.0069	0.0109	0.0155
140.000	0	0.0002	0.0001	0.0049	0.0058	0.0109	0.0165
150.000	0	0.0002	0.0018	0.0040	0.0067	0.0111	0.0196
160.000	0	0.0002	0.0002	0.0039	0.0047	0.0120	0.0182
170.000	0	0.0002	0.0018	0.0038	0.0066	0.0130	0.0183
180.000	0	0.0002	0.0002	0.0042	0.0048	0.0122	0.0172
190.000	0	0.0002	0.0017	0.0036	0.0061	0.0131	0.0175
200.000	0	0.0002	0.0001	0.0041	0.0051	0.0124	0.0159

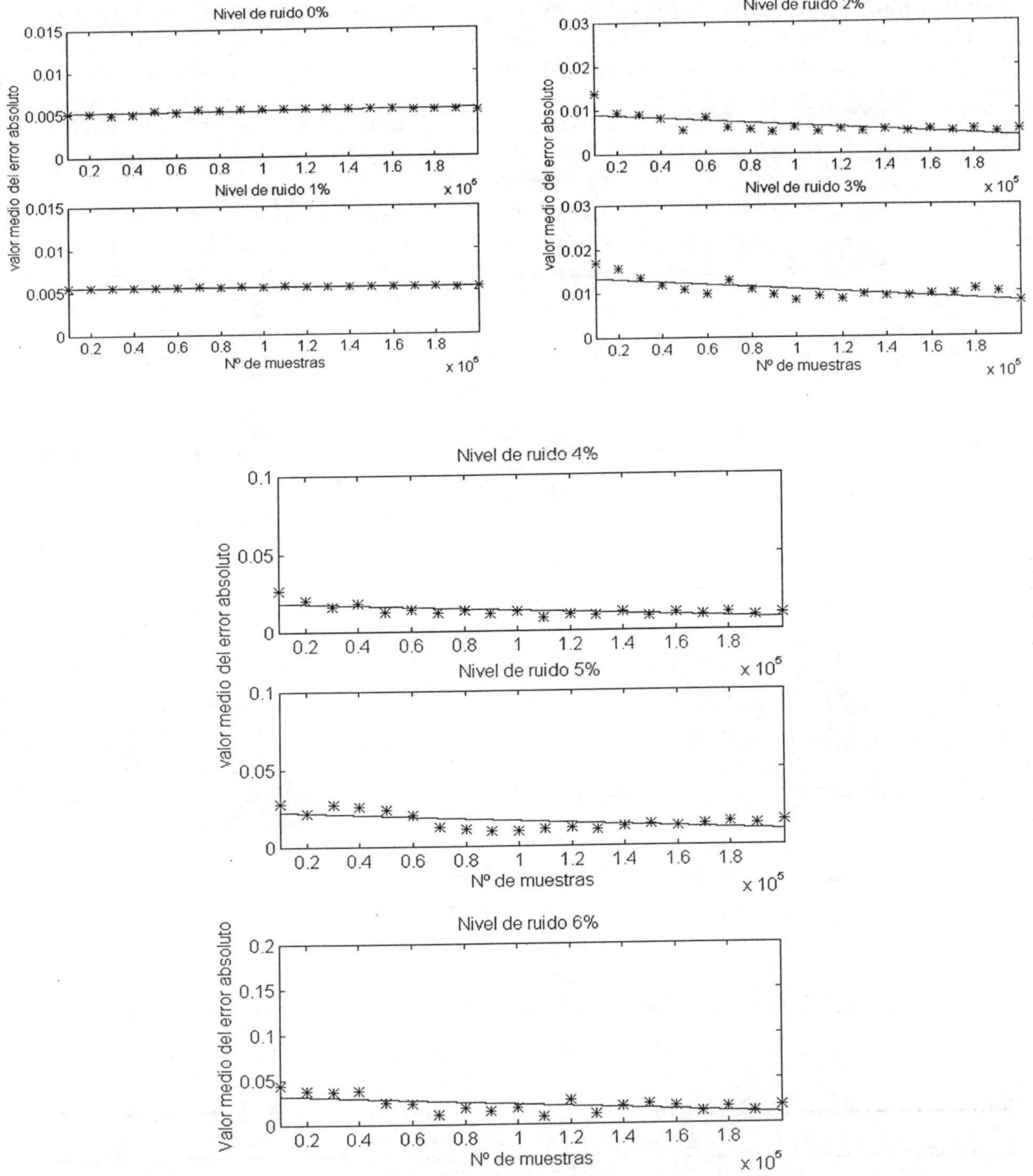


Figura A3.3.3. Valor medio del error absoluto frente al número de muestras para distintos niveles de ruido.

Tabla A3.3.3. Ecuaciones de las rectas de ajuste correspondientes a la Figura A3.3.1, valor del error absoluto para 100.000 muestras y n° de muestras para que el error absoluto sea 0.02.

Nivel de error (%)	Ecuación de la recta	ϵ para $m=100.000$	m_L para $\epsilon = 0.02$
0	$\epsilon = 0.23e-8 m + 0.005$	0.0054	<10.000
1	$\epsilon = - 0.04e-8 m + 0.006$	0.0056	<10.000
2	$\epsilon = - 2.8e-8 m + 0.009$	0.0066	<10.000
3	$\epsilon = - 3e-8 m + 0.014$	0.0109	<10.000
4	$\epsilon = - 5.8e-8 m + 0.020$	0.0132	<10.000
5	$\epsilon = - 6.6e-8 m + 0.023$	0.0165	47175
6	$\epsilon = - 11e-8 m + 0.03$	0.0224	122140

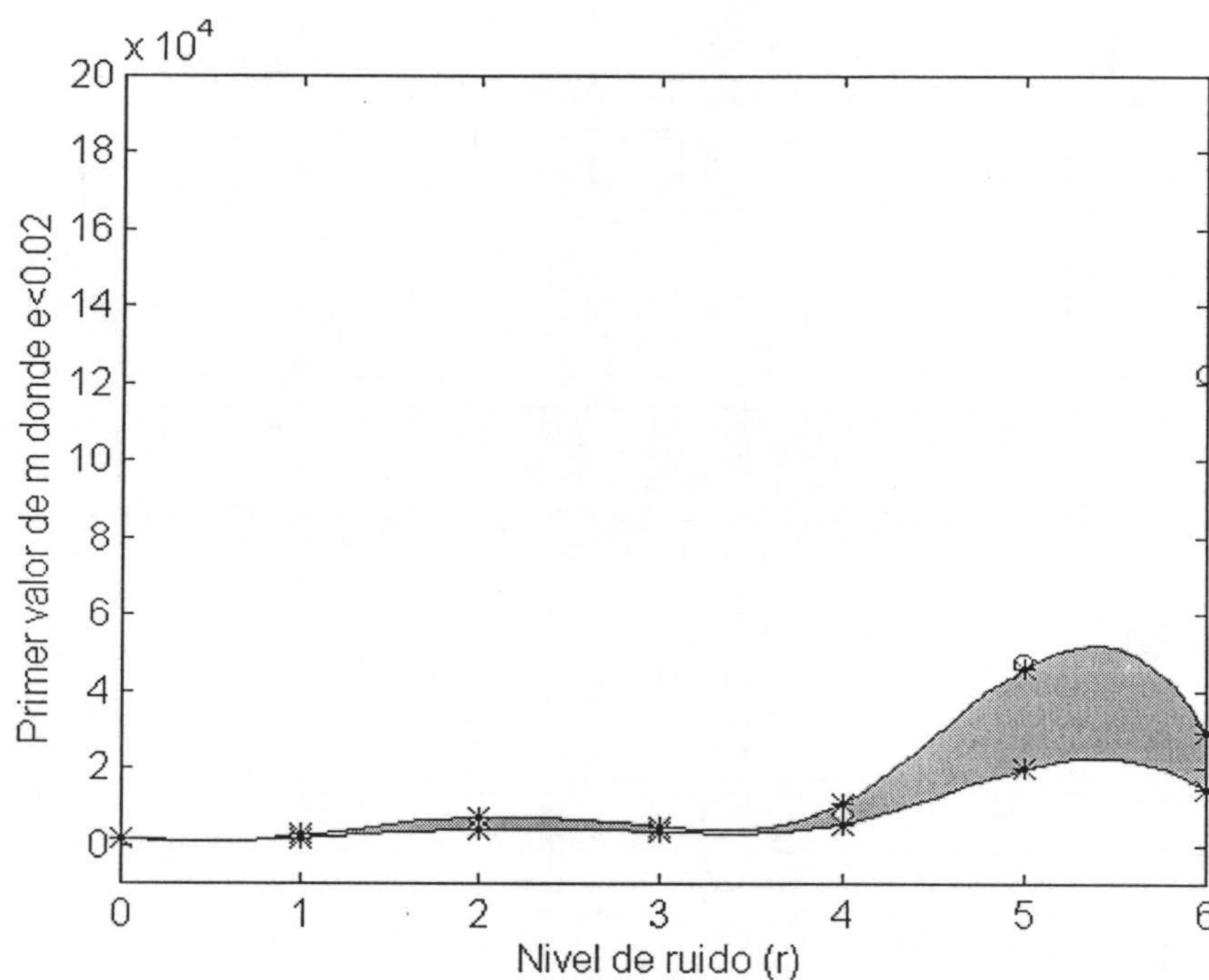


Figura A3.2.4. Primer valor de M donde el error absoluto es menor de 0.02.

Tabla A3.3.4. Primer valor de m con un error absoluto de 0.02, con la matriz A_3 para los distintos niveles de ruido.

Matriz A_3	Nivel de ruido uniforme (%)						
	0	1	2	3	4	5	6
Valor medio	2000	1900	4300	3600	5500	19900	14600
Desviación estándar	0	738	2908	1350	5191	26304	15079

A3.4 RESULTADOS OBTENIDOS CON LA MATRIZ A_4

Tabla A3.4.1. Valor medio del error de estimación de la matriz A_4 (para cada caso se consideran 10 medidas).

m	% Ruido						
	0	1	2	3	4	5	6
10.000	0.0079	0.0095	0.0259	0.1361	0.3712	0.4379	0.5107
20.000	0.0047	0.0091	0.0266	0.1347	0.4198	0.5388	0.5021
30.000	0.0042	0.0082	0.0179	0.2176	0.4375	0.5377	0.5336
40.000	0.0045	0.0079	0.0221	0.2270	0.5021	0.5157	0.5488
50.000	0.0048	0.0069	0.0225	0.2143	0.5208	0.5490	0.5490
60.000	0.0079	0.0092	0.0243	0.2767	0.5049	0.5386	0.5493
70.000	0.0079	0.0069	0.0211	0.2766	0.5160	0.5386	0.5441
80.000	0.0078	0.0069	0.0234	0.2171	0.5423	0.5392	0.5344
90.000	0.0079	0.0069	0.0158	0.3258	0.5373	0.5383	0.5445
100.000	0.0079	0.0081	0.0242	0.3060	0.5316	0.5302	0.5441
110.000	0.0079	0.0079	0.0212	0.2685	0.5338	0.5492	0.5441
120.000	0.0079	0.0079	0.0252	0.3105	0.5339	0.5601	0.5448
130.000	0.0079	0.0079	0.0199	0.2589	0.5535	0.5383	0.5441
140.000	0.0079	0.0054	0.0257	0.2589	0.5330	0.5601	0.5440
150.000	0.0079	0.0054	0.0197	0.1927	0.5471	0.5592	0.5445
160.000	0.0078	0.0079	0.0288	0.1912	0.5579	0.5592	0.5596
170.000	0.0078	0.0079	0.0197	0.1488	0.5636	0.5650	0.5598
180.000	0.0053	0.0066	0.0202	0.1173	0.5526	0.5647	0.5592
190.000	0.0052	0.0054	0.0197	0.1508	0.5527	0.5590	0.5437
200.000	0.0050	0.0054	0.0274	0.1105	0.5526	0.5647	0.5433

Tabla A3.4.2. Desviaciones estándar de los valores medios del error.

m	% Ruido						
	0	1	2	3	4	5	6
10.000	0	0.0047	0.0172	0.1621	0.1306	0.1333	0.0573
20.000	0	0.0049	0.0124	0.1520	0.1396	0.0555	0.1064
30.000	0	0.0052	0.0129	0.2013	0.1465	0.0522	0.0563
40.000	0	0.0053	0.0110	0.2250	0.1239	0.0799	0.0377
50.000	0	0.0040	0.0174	0.2058	0.0537	0.0461	0.0382
60.000	0	0.0061	0.0125	0.2187	0.0864	0.0524	0.0390
70.000	0	0.0040	0.0143	0.2415	0.0875	0.0524	0.0509
80.000	0	0.0040	0.0113	0.2314	0.0469	0.0528	0.0643
90.000	0	0.0040	0.0111	0.2082	0.0531	0.0467	0.0513
100.000	0	0.0052	0.0106	0.2255	0.0518	0.0663	0.0510
110.000	0	0.0053	0.0155	0.2354	0.0671	0.0395	0.0509
120.000	0	0.0053	0.0183	0.2379	0.0671	0.0245	0.0513
130.000	0	0.0053	0.0151	0.2437	0.0368	0.0470	0.0509
140.000	0	0.0001	0.0156	0.2438	0.0670	0.0246	0.0509
150.000	0	0.0001	0.0122	0.2084	0.0386	0.0240	0.0511
160.000	0	0.0053	0.0128	0.2095	0.0241	0.0240	0.0243
170.000	0	0.0053	0.0122	0.1653	0.0182	0.0190	0.0244
180.000	0	0.0040	0.0113	0.1416	0.0372	0.0189	0.0241
190.000	0	0.0001	0.0122	0.1911	0.0372	0.0250	0.0512
200.000	0	0.0001	0.0091	0.1426	0.0372	0.0189	0.0509

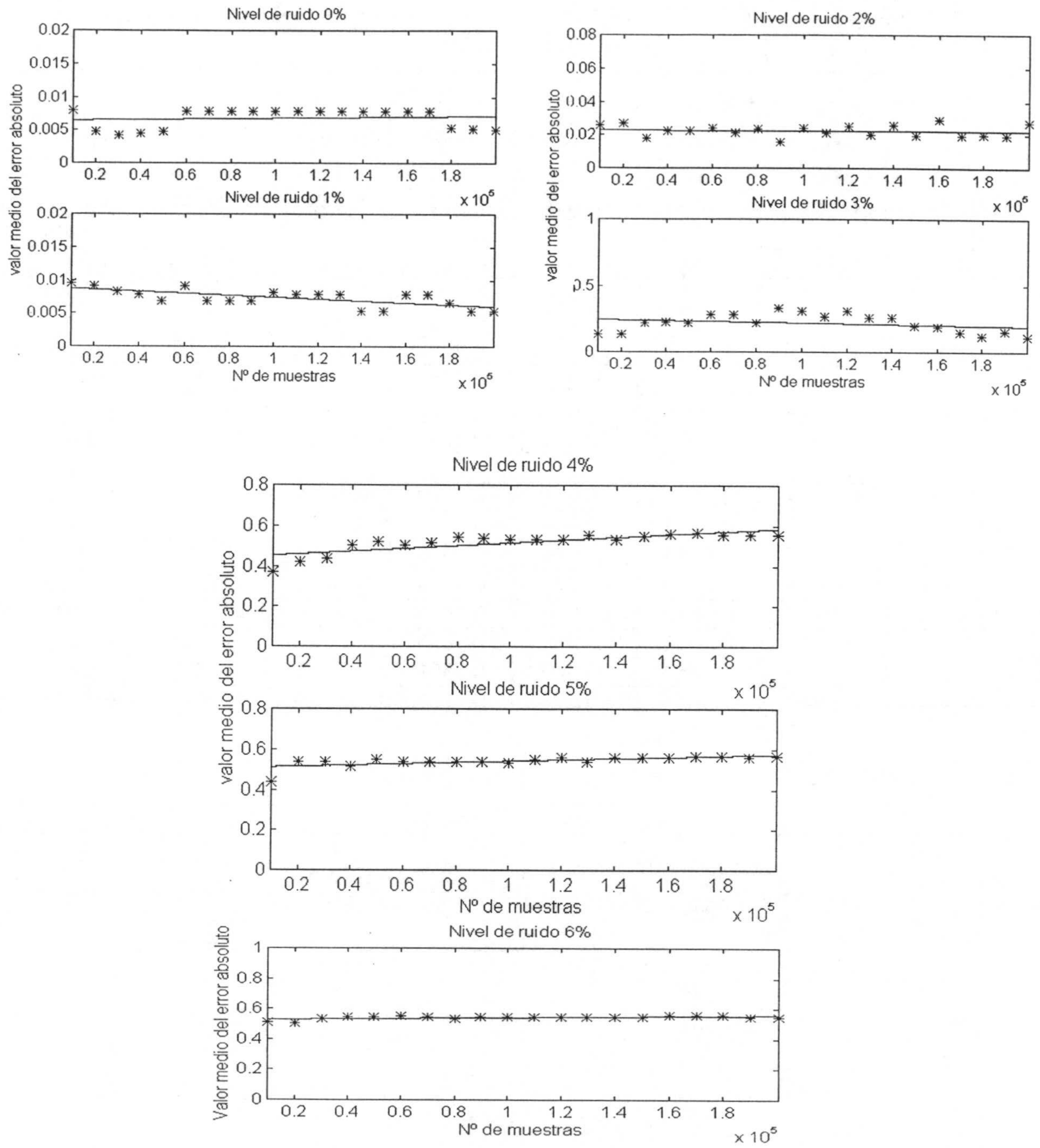


Figura A3.3.5. Valor medio del error absoluto frente al número de muestras para distintos niveles de ruido.

Tabla A3.4.3. Ecuaciones de las rectas de ajuste correspondientes a la Figura A3.4.1, valor del error absoluto para 100.000 muestras y n° de muestras para que el error absoluto sea 0.02.

Nivel de error (%)	Ecuación de la recta	ϵ para $m=100.000$	m_L para $\epsilon = 0.02$
0	$\epsilon = 0.04e-9 m + 0.006$	0.0068	<10.000
1	$\epsilon = -0.14e-9 m + 0.009$	0.0074	<10.000
2	$\epsilon = -1.4e-9 m + 0.023$	0.0226	1883201
3	$\epsilon = -2.8e-9 m + 0.3$	0.2184	805512
4	$\epsilon = 6.9e-9 m + 0.5$	0.5148	<10.000
5	$\epsilon = 3.3e-9 m + 0.5$	0.5405	<10.000
6	$\epsilon = 1.5e-9 m + 0.5$	0.5416	<10.000

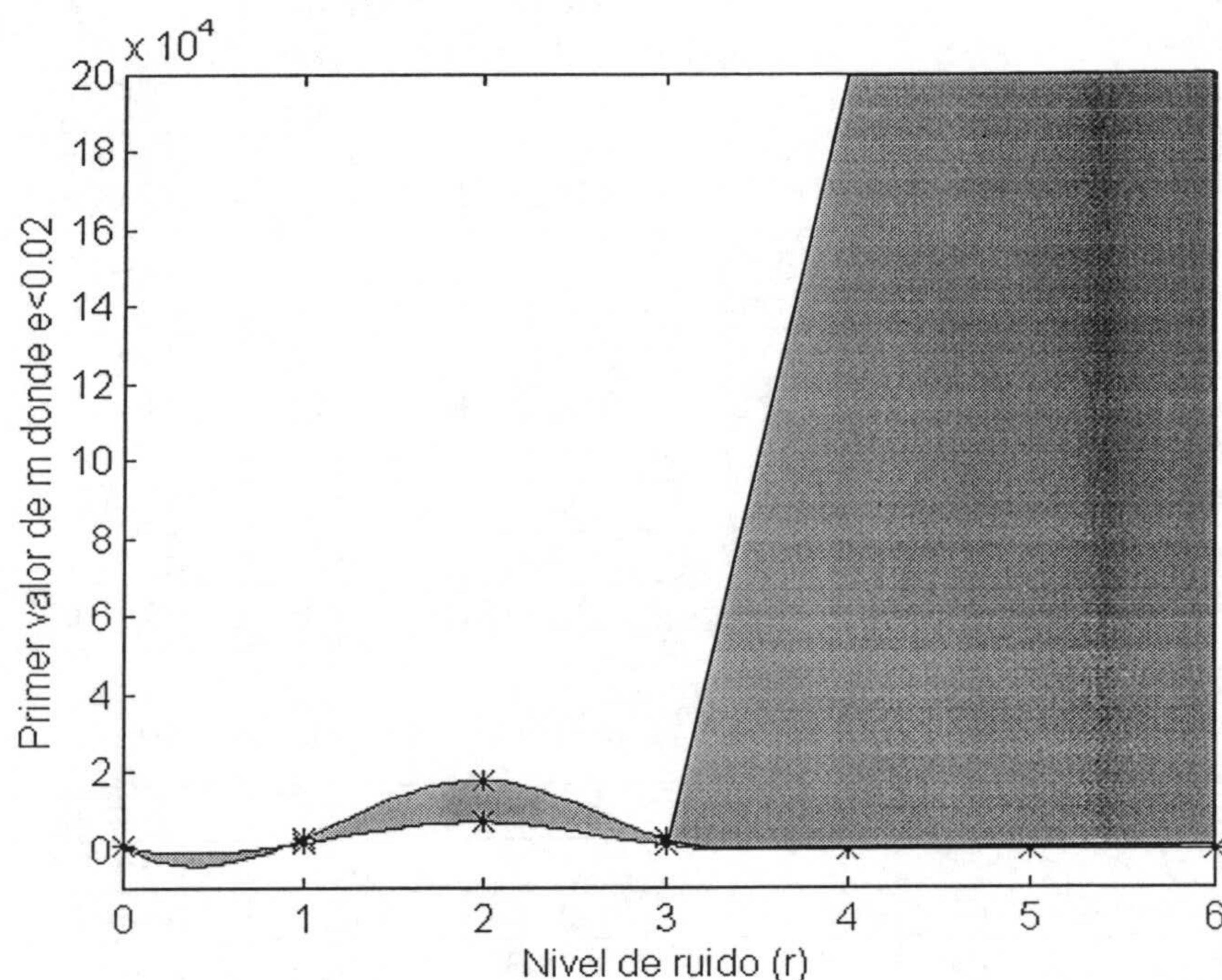


Figura A3.2.6. Primer valor de M donde el error absoluto es menor de 0.02.

Tabla A3.4.4. Primer valor de m con un error absoluto de 0.02, con la matriz A_4 para los distintos niveles de ruido.

	Nivel de ruido uniforme (%)						
Matriz A_4	0	1	2	3	4	5	6
Valor medio	1.000	1.700	7.100	900	>200.000	>200.000	>200.000
Desviación estándar	0	1252	10847	2025	---	---	---

A3.5 RESULTADOS OBTENIDOS CON LA MATRIZ A₅

Tabla A3.5.1. Valor medio del error de estimación de la matriz A₅ (para cada caso se consideran 10 medidas).

m	% Ruido						
	0	1	2	3	4	5	6
10.000	0.0026	0.0032	0.0097	0.0170	0.0187	0.0312	0.0719
20.000	0.0023	0.0044	0.0079	0.0172	0.0178	0.0316	0.0409
30.000	0.0021	0.0039	0.0065	0.0115	0.0105	0.0216	0.0285
40.000	0.0022	0.0040	0.0083	0.0091	0.0184	0.0243	0.0456
50.000	0.0023	0.0037	0.0066	0.0085	0.0103	0.0223	0.0270
60.000	0.0024	0.0036	0.0069	0.0112	0.0137	0.0267	0.0429
70.000	0.0025	0.0037	0.0057	0.0074	0.0113	0.0215	0.0298
80.000	0.0025	0.0035	0.0070	0.0106	0.0147	0.0273	0.0281
90.000	0.0025	0.0035	0.0056	0.0080	0.0109	0.0259	0.0263
100.000	0.0026	0.0037	0.0055	0.0094	0.0144	0.0284	0.0340
110.000	0.0026	0.0033	0.0060	0.0072	0.0131	0.0228	0.0237
120.000	0.0026	0.0033	0.0057	0.0092	0.0131	0.0268	0.0326
130.000	0.0026	0.0034	0.0064	0.0065	0.0131	0.0215	0.0205
140.000	0.0035	0.0033	0.0054	0.0087	0.0134	0.0243	0.0292
150.000	0.0035	0.0032	0.0051	0.0084	0.0116	0.0177	0.0207
160.000	0.0026	0.0032	0.0046	0.0082	0.0127	0.0192	0.0294
170.000	0.0025	0.0029	0.0051	0.0074	0.0125	0.0181	0.0196
180.000	0.0025	0.0029	0.0048	0.0079	0.0122	0.0205	0.0241
190.000	0.0025	0.0030	0.0047	0.0060	0.0125	0.0168	0.0207
200.000	0.0024	0.0030	0.0050	0.0075	0.0137	0.0191	0.0256

Tabla A3.5.2. Desviaciones estándar de los valores medios del error.

m	% Ruido						
	0	1	2	3	4	5	6
10.000	0	0.0011	0.0074	0.0080	0.0122	0.0152	0.0740
20.000	0	0.0018	0.0032	0.0093	0.0122	0.0237	0.0171
30.000	0	0.0017	0.0031	0.0101	0.0075	0.0152	0.0230
40.000	0	0.0017	0.0038	0.0074	0.0110	0.0123	0.0396
50.000	0	0.0018	0.0028	0.0068	0.0078	0.0118	0.0217
60.000	0	0.0018	0.0024	0.0065	0.0068	0.0149	0.0150
70.000	0	0.0018	0.0023	0.0068	0.0095	0.0135	0.0223
80.000	0	0.0018	0.0041	0.0062	0.0082	0.0109	0.0130
90.000	0	0.0018	0.0027	0.0058	0.0088	0.0145	0.0228
100.000	0	0.0018	0.0027	0.0069	0.0073	0.0095	0.0178
110.000	0	0.0014	0.0033	0.0060	0.0097	0.0149	0.0193
120.000	0	0.0014	0.0022	0.0046	0.0079	0.0116	0.0162
130.000	0	0.0013	0.0035	0.0053	0.0099	0.0144	0.0165
140.000	0	0.0014	0.0016	0.0056	0.0080	0.0113	0.0132
150.000	0	0.0014	0.0017	0.0056	0.0086	0.0129	0.0165
160.000	0	0.0014	0.0017	0.0053	0.0072	0.0115	0.0138
170.000	0	0.0006	0.0023	0.0061	0.0085	0.0123	0.0164
180.000	0	0.0006	0.0017	0.0048	0.0057	0.0106	0.0122
190.000	0	0.0006	0.0018	0.0052	0.0085	0.0110	0.0164
200.000	0	0.0006	0.0023	0.0051	0.0069	0.0098	0.0105

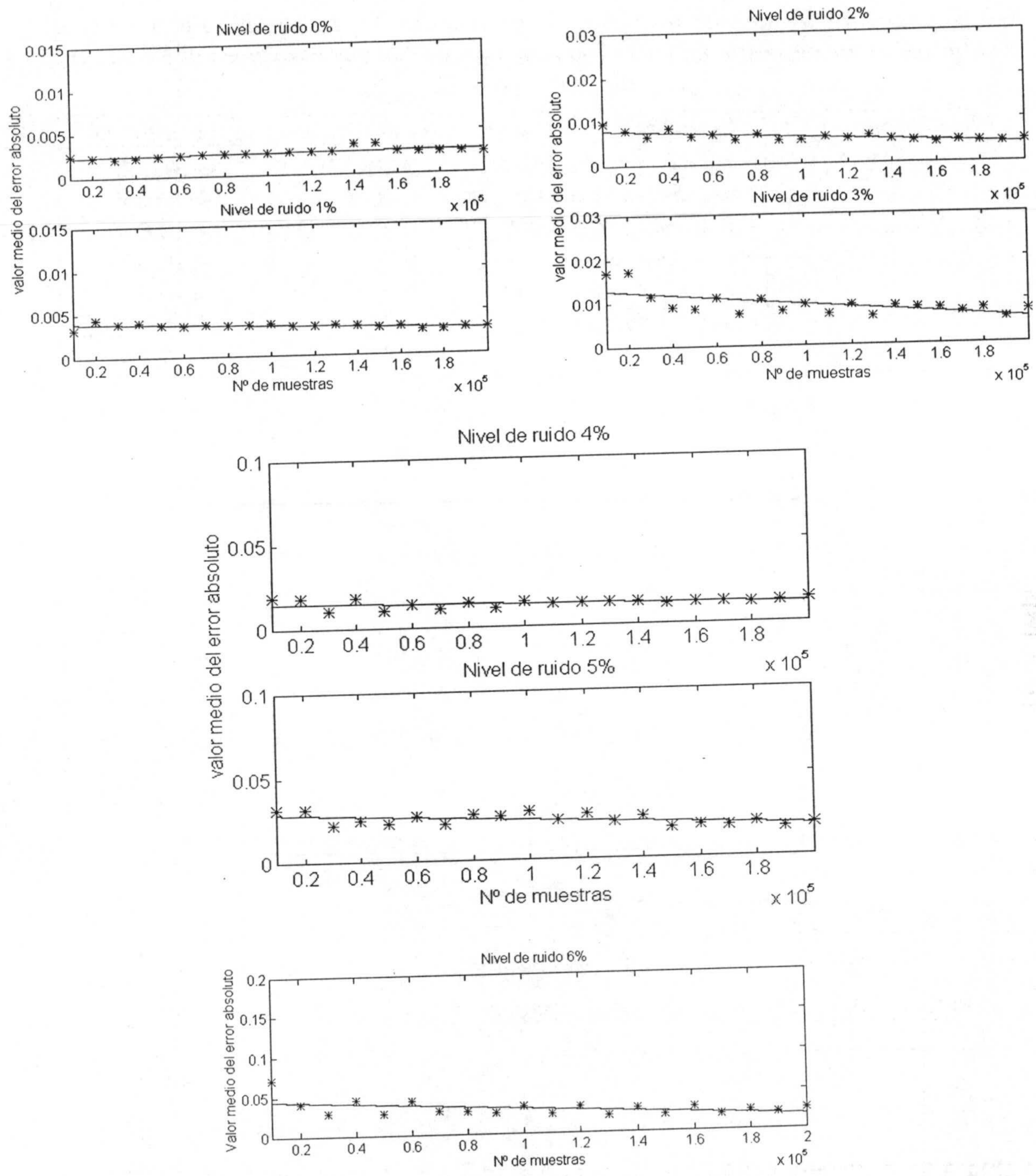


Figura A3.5.1. Valor medio del error absoluto frente al número de muestras para distintos niveles de ruido.

Tabla A3.5.3. Ecuaciones de las rectas de ajuste correspondientes a la Figura A3.5.1, valor del error absoluto para 100.000 muestras y n° de muestras para que el error absoluto sea 0.02.

Nivel de error (%)	Ecuación de la recta	ϵ para $m=100.000$	m_L para $\epsilon = 0.02$
0	$\epsilon = 0.23e-8 m + 0.0023$	0.0025	< 10.000
1	$\epsilon = -0.54e-8 m + 0.004$	0.0035	< 10.000
2	$\epsilon = -0.19e-8 m + 0.008$	0.0062	< 10.000
3	$\epsilon = -3.7e-8 m + 0.013$	0.0095	< 10.000
4	$\epsilon = -1.6e-8 m + 0.015$	0.0135	< 10.000
5	$\epsilon = -5.3e-8 m + 0.03$	0.0236	168.716
6	$\epsilon = -1.4e-7 m + 0.05$	0.0317	185.265

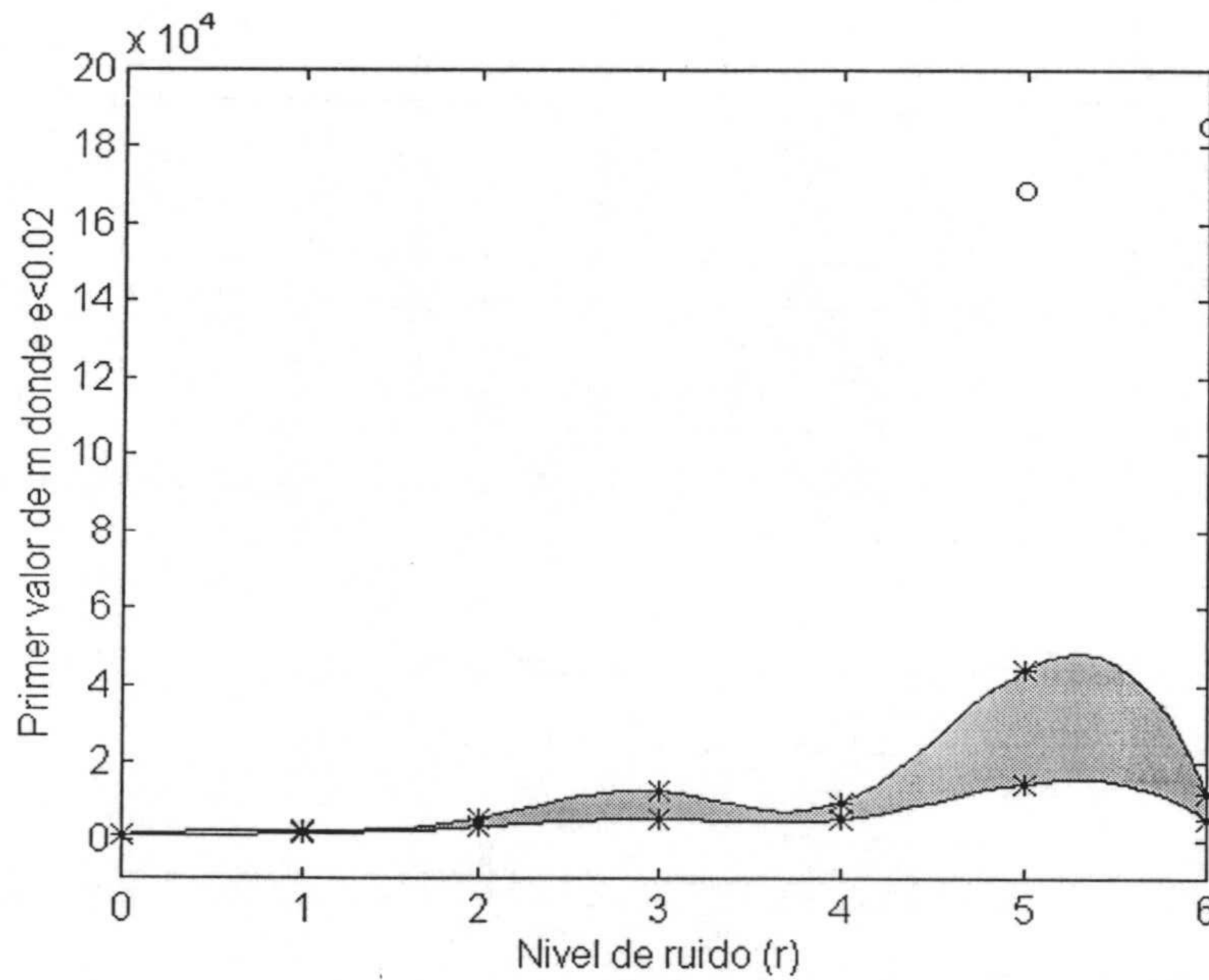


Figura A3.5.2. Primer valor de M donde el error absoluto es menor de 0.02.

Tabla A3.5.4. Primer valor de m con un error absoluto de 0.02, con la matriz A_5 para los distintos niveles de ruido.

Matriz A_5	Nivel de ruido uniforme (%)						
	0	1	2	3	4	5	6
Valor medio	1000	1600	3300	5500	5400	14900	5300
Desviación estándar	0	699	2312	7307	4326	29309	7009

A3.6 RESULTADOS OBTENIDOS CON LA MATRIZ A₆

Tabla A3.6.1. Valor medio del error de estimación de la matriz A₆ (para cada caso se consideran 10 medidas).

m	% Ruido						
	0	1	2	3	4	5	6
10.000	0.0018	0.0045	0.0085	0.0150	0.0154	0.0307	0.0712
20.000	0.0017	0.0032	0.0084	0.0121	0.0203	0.0279	0.0581
30.000	0.0017	0.0036	0.0067	0.0144	0.0232	0.0203	0.0540
40.000	0.0018	0.0034	0.0089	0.0129	0.0199	0.0223	0.0446
50.000	0.0018	0.0033	0.0078	0.0113	0.0172	0.0151	0.0409
60.000	0.0018	0.0030	0.0071	0.0117	0.0175	0.0265	0.0399
70.000	0.0018	0.0025	0.0069	0.0114	0.0176	0.0191	0.0278
80.000	0.0018	0.0020	0.0067	0.0107	0.0254	0.0242	0.0331
90.000	0.0018	0.0020	0.0079	0.0101	0.0233	0.0210	0.0262
100.000	0.0019	0.0020	0.0069	0.0105	0.0233	0.0219	0.0304
110.000	0.0019	0.0020	0.0060	0.0097	0.0175	0.0191	0.0226
120.000	0.0019	0.0025	0.0060	0.0088	0.0181	0.0240	0.0326
130.000	0.0019	0.0020	0.0062	0.0097	0.0165	0.0182	0.0249
140.000	0.0019	0.0020	0.0067	0.0088	0.0199	0.0221	0.0289
150.000	0.0019	0.0020	0.0067	0.0113	0.0147	0.0209	0.0286
160.000	0.0018	0.0020	0.0059	0.0104	0.0189	0.0234	0.0315
170.000	0.0018	0.0020	0.0057	0.0106	0.0166	0.0176	0.0292
180.000	0.0018	0.0021	0.0060	0.0104	0.0183	0.0246	0.0358
190.000	0.0018	0.0020	0.0069	0.0115	0.0143	0.0164	0.0272
200.000	0.0017	0.0021	0.0077	0.0122	0.0175	0.0199	0.0378

Tabla A3.6.2. Desviaciones estándar de los valores medios del error.

m	% Ruido						
	0	1	2	3	4	5	6
10.000	0	0.0020	0.0044	0.0091	0.0119	0.0147	0.0445
20.000	0	0.0020	0.0038	0.0064	0.0058	0.0095	0.0274
30.000	0	0.0022	0.0030	0.0074	0.0157	0.0154	0.0349
40.000	0	0.0023	0.0030	0.0055	0.0130	0.0156	0.0188
50.000	0	0.0021	0.0029	0.0054	0.0108	0.0141	0.0272
60.000	0	0.0021	0.0023	0.0046	0.0108	0.0091	0.0249
70.000	0	0.0015	0.0026	0.0046	0.0117	0.0152	0.0261
80.000	0	0.0000	0.0039	0.0043	0.0072	0.0133	0.0232
90.000	0	0.0000	0.0032	0.0047	0.0108	0.0144	0.0244
100.000	0	0.0000	0.0031	0.0035	0.0065	0.0122	0.0191
110.000	0	0.0000	0.0030	0.0035	0.0069	0.0155	0.0211
120.000	0	0.0015	0.0030	0.0024	0.0082	0.0068	0.0150
130.000	0	0.0000	0.0027	0.0036	0.0096	0.0144	0.0219
140.000	0	0.0000	0.0028	0.0024	0.0115	0.0105	0.0225
150.000	0	0.0000	0.0028	0.0043	0.0092	0.0127	0.0218
160.000	0	0.0000	0.0026	0.0052	0.0095	0.0120	0.0204
170.000	0	0.0000	0.0027	0.0047	0.0098	0.0148	0.0265
180.000	0	0.0000	0.0034	0.0051	0.0088	0.0079	0.0211
190.000	0	0.0000	0.0032	0.0049	0.0096	0.0140	0.0241
200.000	0	0.0000	0.0022	0.0043	0.0085	0.0109	0.0163

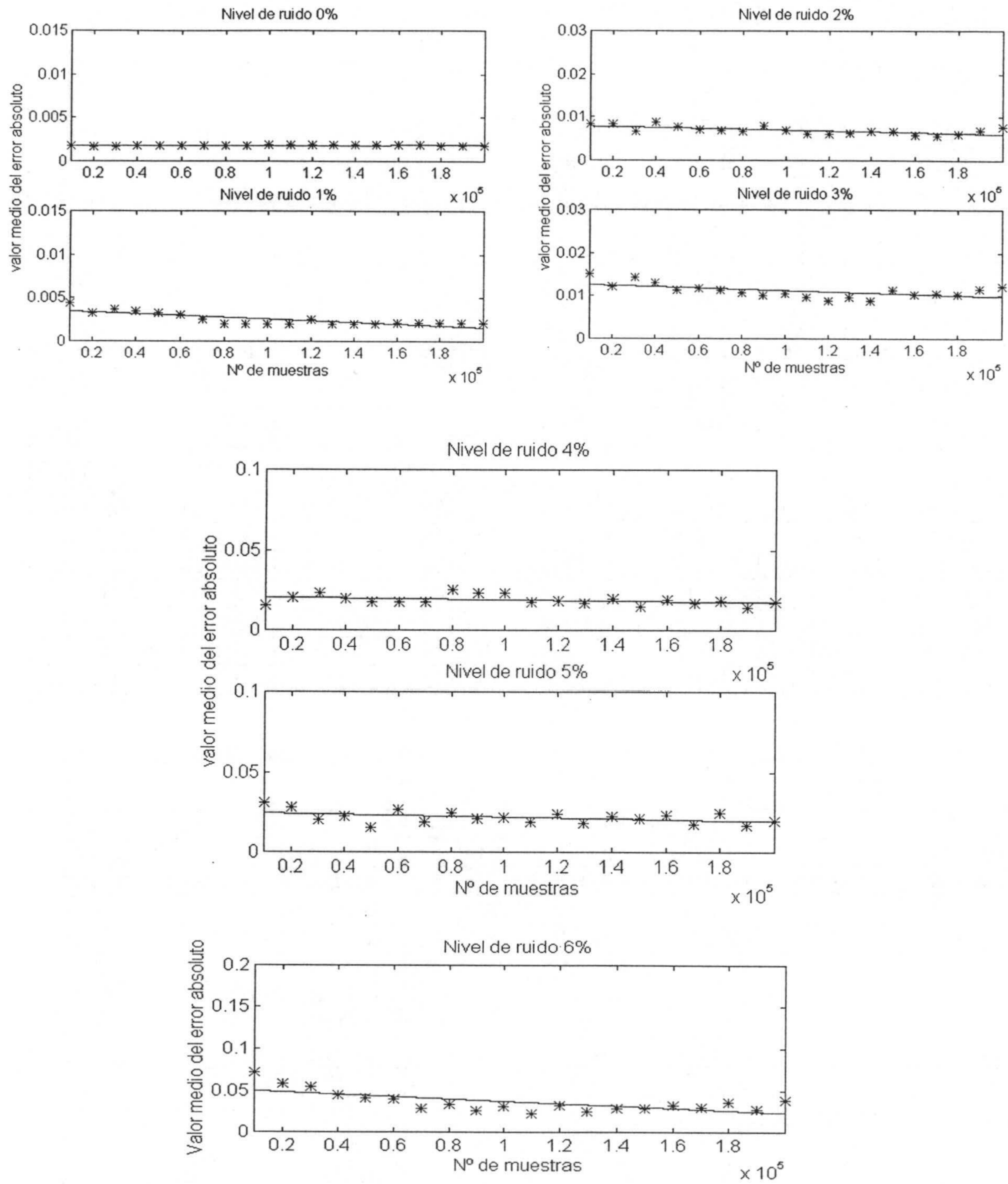


Figura A3.6.1. Valor medio del error absoluto frente al número de muestras para distintos niveles de ruido.

Tabla A3.6.3. Ecuaciones de las rectas de ajuste correspondientes a la Figura A3.6.1, valor del error absoluto para 100.000 muestras y n° de muestras para que el error absoluto sea 0.02.

Nivel de error (%)	Ecuación de la recta	ϵ para $m=100.000$	m_L para $\epsilon = 0.02$
0	$\epsilon = 0.05e-8 m + 0.0018$	0.0018	<10.000
1	$\epsilon = - 1.0e-8 m + 0.004$	0.0026	<10.000
2	$\epsilon = - 1.0e-8 m + 0.008$	0.0070	<10.000
3	$\epsilon = - 1.5e-8 m + 0.013$	0.0112	<10.000
4	$\epsilon = - 1.8e-8 m + 0.021$	0.0189	34737
5	$\epsilon = - 2.7e-8 m + 0.03$	0.0219	170573
6	$\epsilon = - 14.0e-8 m + 0.05$	0.0370	221107

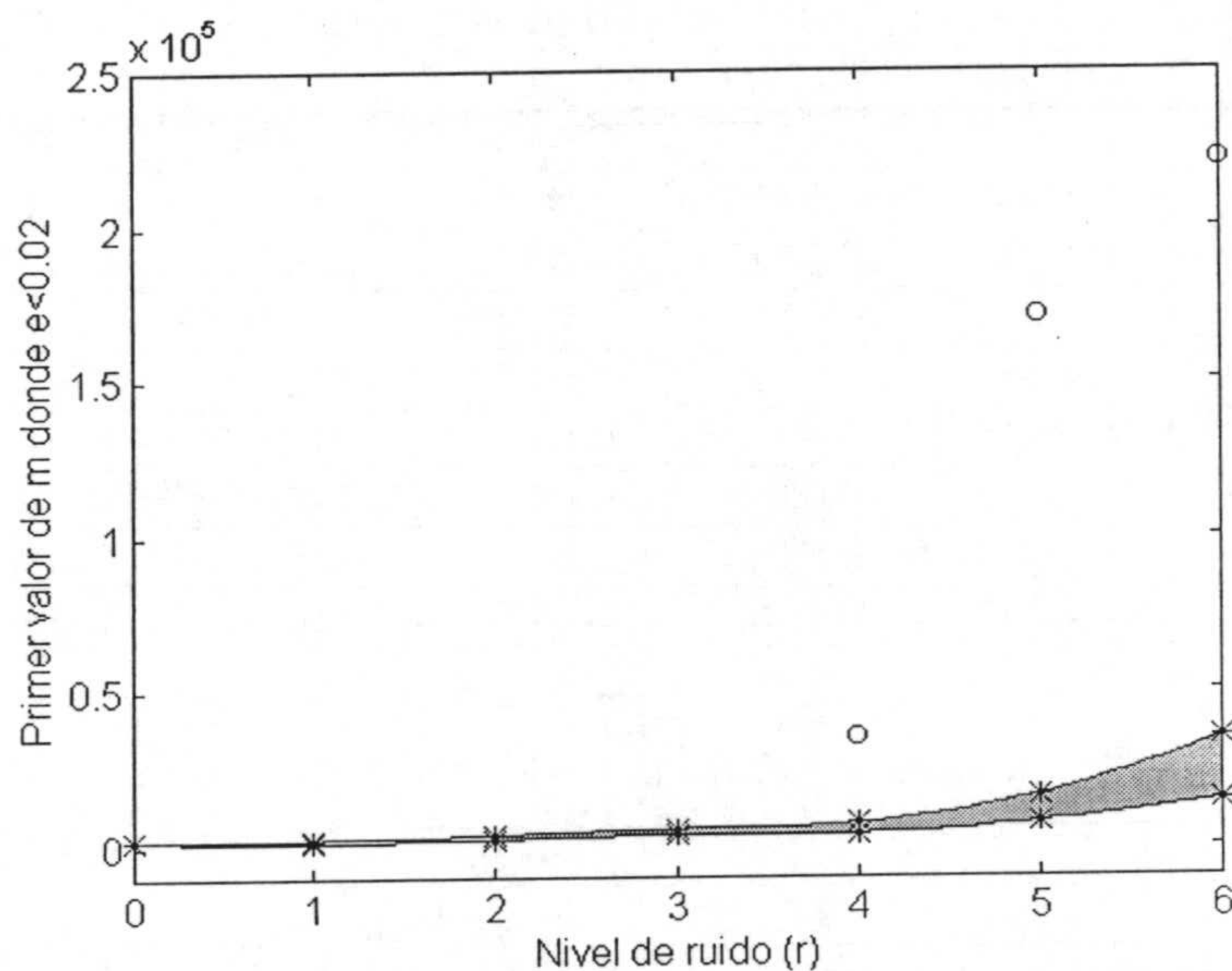


Figura A3.2.7. Primer valor de M donde el error absoluto es menor de 0.02.

Tabla A3.6.4. Primer valor de m con un error absoluto de 0.02, con la matriz A_6 para los distintos niveles de ruido.

Matriz A_6	Nivel de ruido uniforme (%)						
	0	1	2	3	4	5	6
Valor medio	2000	1600	2400	3700	4000	7400	14600
Desviación estándar	0	966	1350	1889	3232	8847	20299

A3.7 RESULTADOS OBTENIDOS CON LA MATRIZ A₇

Tabla A3.7.1. Valor medio del error de estimación de la matriz A₇ (para cada caso se consideran 10 medidas).

m	% Ruido						
	0	1	2	3	4	5	6
10.000	0.0054	0.0077	0.0176	0.0261	0.0432	0.1233	0.1560
20.000	0.0007	0.0049	0.0152	0.0263	0.0315	0.0712	0.1976
30.000	0.0005	0.0054	0.0117	0.0211	0.0189	0.0700	0.1976
40.000	0.0005	0.0048	0.0094	0.0198	0.0267	0.0557	0.2602
50.000	0.0005	0.0059	0.0095	0.0134	0.0172	0.0498	0.2467
60.000	0.0054	0.0054	0.0080	0.0184	0.0265	0.0482	0.1581
70.000	0.0054	0.0053	0.0096	0.0138	0.0171	0.0385	0.1674
80.000	0.0054	0.0048	0.0069	0.0204	0.0251	0.0425	0.1330
90.000	0.0053	0.0046	0.0119	0.0154	0.0171	0.0293	0.1421
100.000	0.0053	0.0048	0.0109	0.0194	0.0228	0.0329	0.1614
110.000	0.0053	0.0042	0.0092	0.0132	0.0185	0.0185	0.1878
120.000	0.0053	0.0042	0.0112	0.0124	0.0290	0.0312	0.1834
130.000	0.0053	0.0042	0.0110	0.0096	0.0195	0.0159	0.0983
140.000	0.0053	0.0037	0.0117	0.0108	0.0232	0.0253	0.1064
150.000	0.0053	0.0037	0.0133	0.0095	0.0205	0.0264	0.0911
160.000	0.0053	0.0042	0.0117	0.0152	0.0331	0.0337	0.0961
170.000	0.0053	0.0032	0.0100	0.0111	0.0247	0.0160	0.0750
180.000	0.0053	0.0037	0.0133	0.0151	0.0339	0.0313	0.1324
190.000	0.0053	0.0032	0.0106	0.0110	0.0209	0.0160	0.0962
200.000	0.0006	0.0032	0.0102	0.0100	0.0361	0.0309	0.1130

Tabla A3.7. 2. Desviaciones estándar de los valores medios del error.

m	% Ruido						
	0	1	2	3	4	5	6
10.000	0	0.0068	0.0104	0.0137	0.0202	0.0985	0.1104
20.000	0	0.0055	0.0108	0.0161	0.0166	0.0566	0.1618
30.000	0	0.0047	0.0085	0.0188	0.0185	0.0679	0.1307
40.000	0	0.0032	0.0064	0.0149	0.0189	0.0398	0.1606
50.000	0	0.0034	0.0071	0.0072	0.0188	0.0472	0.1974
60.000	0	0.0029	0.0052	0.0075	0.0203	0.0363	0.1050
70.000	0	0.0029	0.0084	0.0096	0.0188	0.0440	0.1922
80.000	0	0.0033	0.0069	0.0108	0.0190	0.0292	0.1255
90.000	0	0.0032	0.0086	0.0090	0.0188	0.0308	0.1147
100.000	0	0.0032	0.0115	0.0096	0.0147	0.0298	0.1300
110.000	0	0.0034	0.0088	0.0077	0.0198	0.0285	0.1649
120.000	0	0.0034	0.0051	0.0099	0.0188	0.0328	0.1504
130.000	0	0.0034	0.0052	0.0095	0.0202	0.0309	0.0931
140.000	0	0.0035	0.0049	0.0107	0.0138	0.0319	0.1119
150.000	0	0.0035	0.0029	0.0078	0.0173	0.0369	0.0585
160.000	0	0.0034	0.0049	0.0101	0.0091	0.0322	0.0936
170.000	0	0.0036	0.0056	0.0074	0.0191	0.0265	0.0603
180.000	0	0.0036	0.0029	0.0087	0.0142	0.0279	0.1279
190.000	0	0.0036	0.0060	0.0059	0.0170	0.0265	0.0776
200.000	0	0.0036	0.0061	0.0057	0.0159	0.0292	0.1042

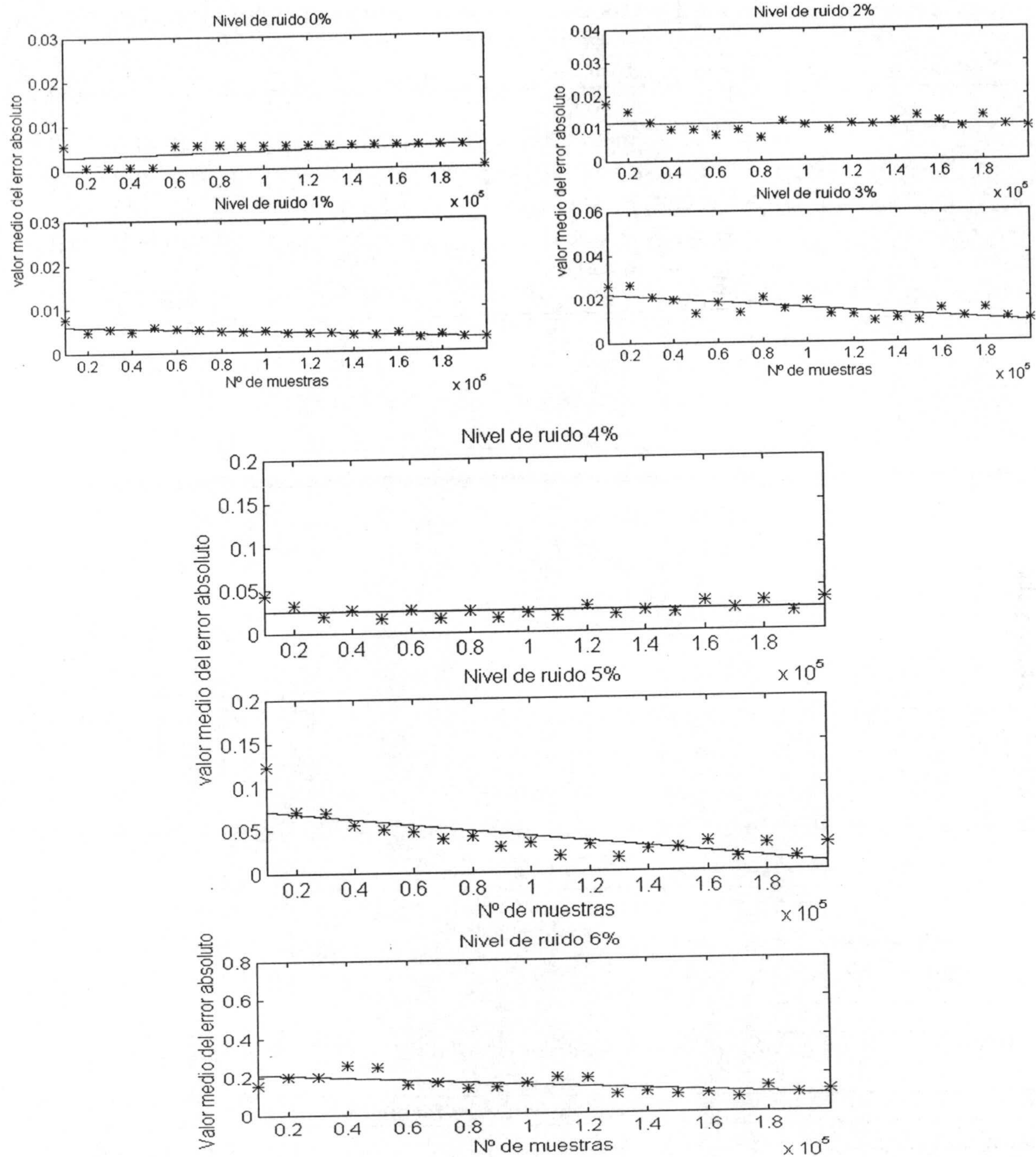


Figura A3.7.1. Valor medio del error absoluto frente al número de muestras para distintos niveles de ruido.

Tabla A3.7.3. Ecuaciones de las rectas de ajuste correspondientes a la Figura A3.7.1, valor del error absoluto para 100.000 muestras y n° de muestras para que el error absoluto sea 0.02.

Nivel de error (%)	Ecuación de la recta	ϵ para $m=100.000$	m_L para $\epsilon = 0.02$
0	$\epsilon = 1.3e-8 m + 0.003$	0.0041	<10.000
1	$\epsilon = - 1.6e-8 m + 0.006$	0.0046	<10.000
2	$\epsilon = - 0.64e-8 m + 0.011$	0.0112	<10.000
3	$\epsilon = - 6.9e-8 m + 0.023$	0.0160	41374
4	$\epsilon = 0.38e-8 m + 0.03$	0.0253	<10.000
5	$\epsilon = - 33e-8 m + 0.08$	0.0420	166465
6	$\epsilon = - 65e-8 m + 0.022$	0.1532	305221

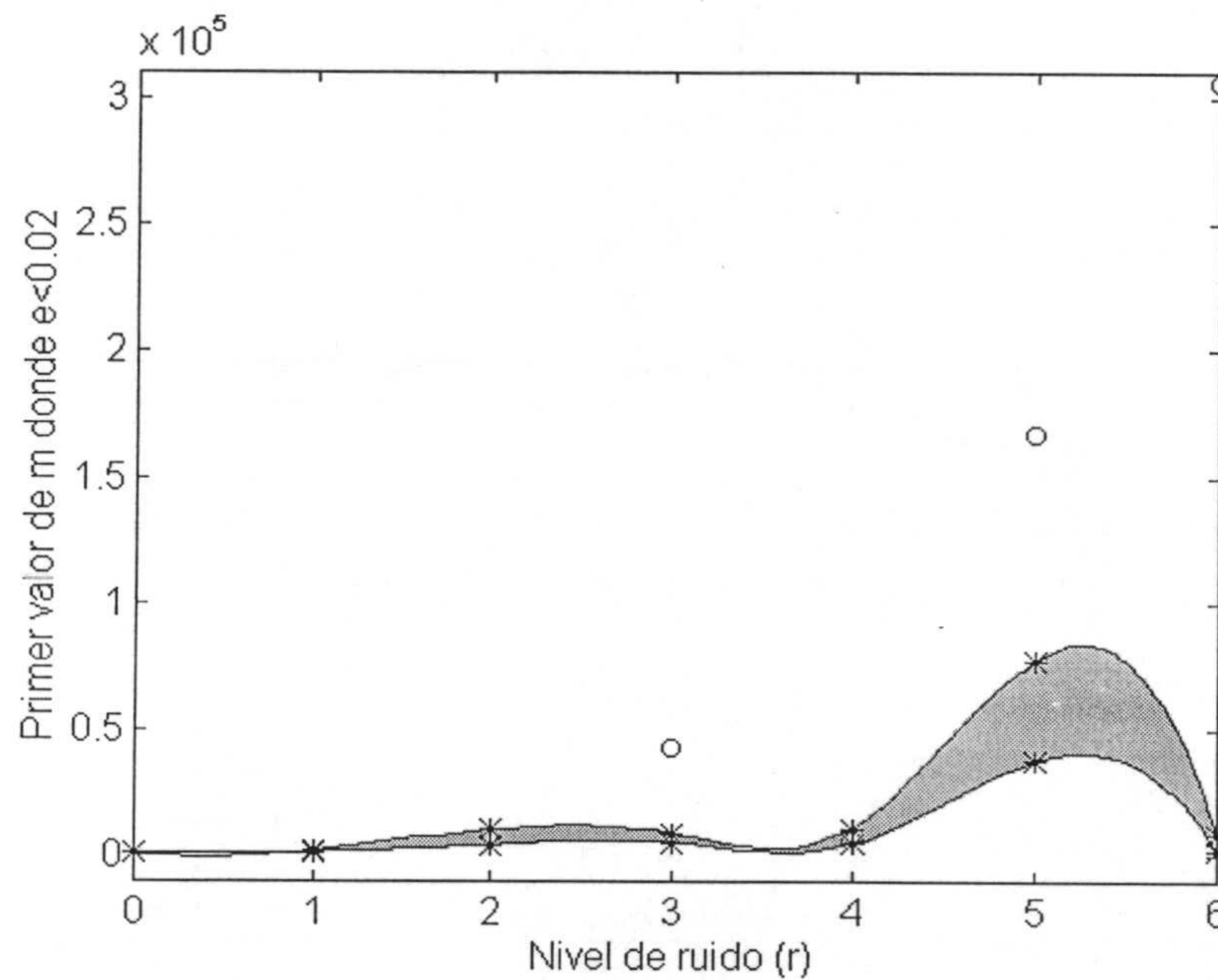


Figura A3.7.2. Primer valor de M donde el error absoluto es menor de 0.02.

Tabla A3.7.4. Primer valor de m con un error absoluto de 0.02, con la matriz A_7 para los distintos niveles de ruido.

	Nivel de ruido uniforme (%)						
Matriz A_7	0	1	2	3	4	5	6
Valor medio	1000	1600	4400	5300	4900	37900	2500
Desviación estándar	0	699	6096	3917	6173	39776	6241

A3.8 RESULTADOS OBTENIDOS CON LA MATRIZ A_8

Tabla A3.8.1. Valor medio del error de estimación de la matriz A_8 (para cada caso se consideran 10 medidas).

m	% Ruido						
	0	1	2	3	4	5	6
10.000	0.0058	0.0088	0.0127	0.0134	0.0099	0.0162	0.0160
20.000	0.0038	0.0069	0.0079	0.0187	0.0123	0.0121	0.0128
30.000	0.0038	0.0065	0.0085	0.0164	0.0137	0.0077	0.0050
40.000	0.0039	0.0061	0.0068	0.0120	0.0169	0.0170	0.0140
50.000	0.0040	0.0060	0.0063	0.0095	0.0127	0.0109	0.0041
60.000	0.0042	0.0052	0.0075	0.0121	0.0125	0.0129	0.0130
70.000	0.0042	0.0050	0.0060	0.0084	0.0080	0.0057	0.0060
80.000	0.0042	0.0054	0.0072	0.0082	0.0140	0.0134	0.0118
90.000	0.0042	0.0052	0.0061	0.0090	0.0090	0.0086	0.0070
100.000	0.0042	0.0055	0.0054	0.0089	0.0121	0.0165	0.0114
110.000	0.0042	0.0054	0.0061	0.0089	0.0080	0.0075	0.0053
120.000	0.0042	0.0054	0.0053	0.0093	0.0121	0.0178	0.0116
130.000	0.0042	0.0054	0.0049	0.0067	0.0099	0.0102	0.0075
140.000	0.0042	0.0052	0.0053	0.0083	0.0106	0.0146	0.0119
150.000	0.0041	0.0055	0.0049	0.0071	0.0069	0.0097	0.0082
160.000	0.0041	0.0053	0.0054	0.0080	0.0085	0.0149	0.0120
170.000	0.0041	0.0053	0.0055	0.0063	0.0067	0.0084	0.0068
180.000	0.0041	0.0053	0.0066	0.0078	0.0082	0.0151	0.0126
190.000	0.0041	0.0053	0.0055	0.0060	0.0067	0.0088	0.0053
200.000	0.0040	0.0053	0.0066	0.0088	0.0111	0.0172	0.0105

Tabla A3.8.2. Desviaciones estándar de los valores medios del error.

m	% Ruido						
	0	1	2	3	4	5	6
10.000	0	0.0050	0.0081	0.0105	0.0100	0.0237	0.0232
20.000	0	0.0034	0.0035	0.0165	0.0097	0.0092	0.0150
30.000	0	0.0015	0.0049	0.0134	0.0134	0.0114	0.0073
40.000	0	0.0017	0.0026	0.0094	0.0145	0.0132	0.0115
50.000	0	0.0018	0.0040	0.0082	0.0141	0.0125	0.0066
60.000	0	0.0009	0.0037	0.0056	0.0103	0.0105	0.0120
70.000	0	0.0008	0.0035	0.0081	0.0082	0.0086	0.0088
80.000	0	0.0009	0.0039	0.0060	0.0124	0.0105	0.0111
90.000	0	0.0009	0.0039	0.0069	0.0093	0.0135	0.0084
100.000	0	0.0009	0.0019	0.0070	0.0131	0.0099	0.0103
110.000	0	0.0009	0.0038	0.0089	0.0082	0.0086	0.0095
120.000	0	0.0009	0.0024	0.0072	0.0075	0.0110	0.0079
130.000	0	0.0009	0.0029	0.0066	0.0079	0.0141	0.0121
140.000	0	0.0009	0.0024	0.0059	0.0091	0.0120	0.0127
150.000	0	0.0009	0.0029	0.0063	0.0063	0.0126	0.0141
160.000	0	0.0009	0.0024	0.0063	0.0062	0.0111	0.0108
170.000	0	0.0009	0.0031	0.0069	0.0064	0.0118	0.0119
180.000	0	0.0009	0.0032	0.0057	0.0064	0.0106	0.0080
190.000	0	0.0009	0.0031	0.0063	0.0064	0.0126	0.0087
200.000	0	0.0009	0.0032	0.0061	0.0067	0.0104	0.0093

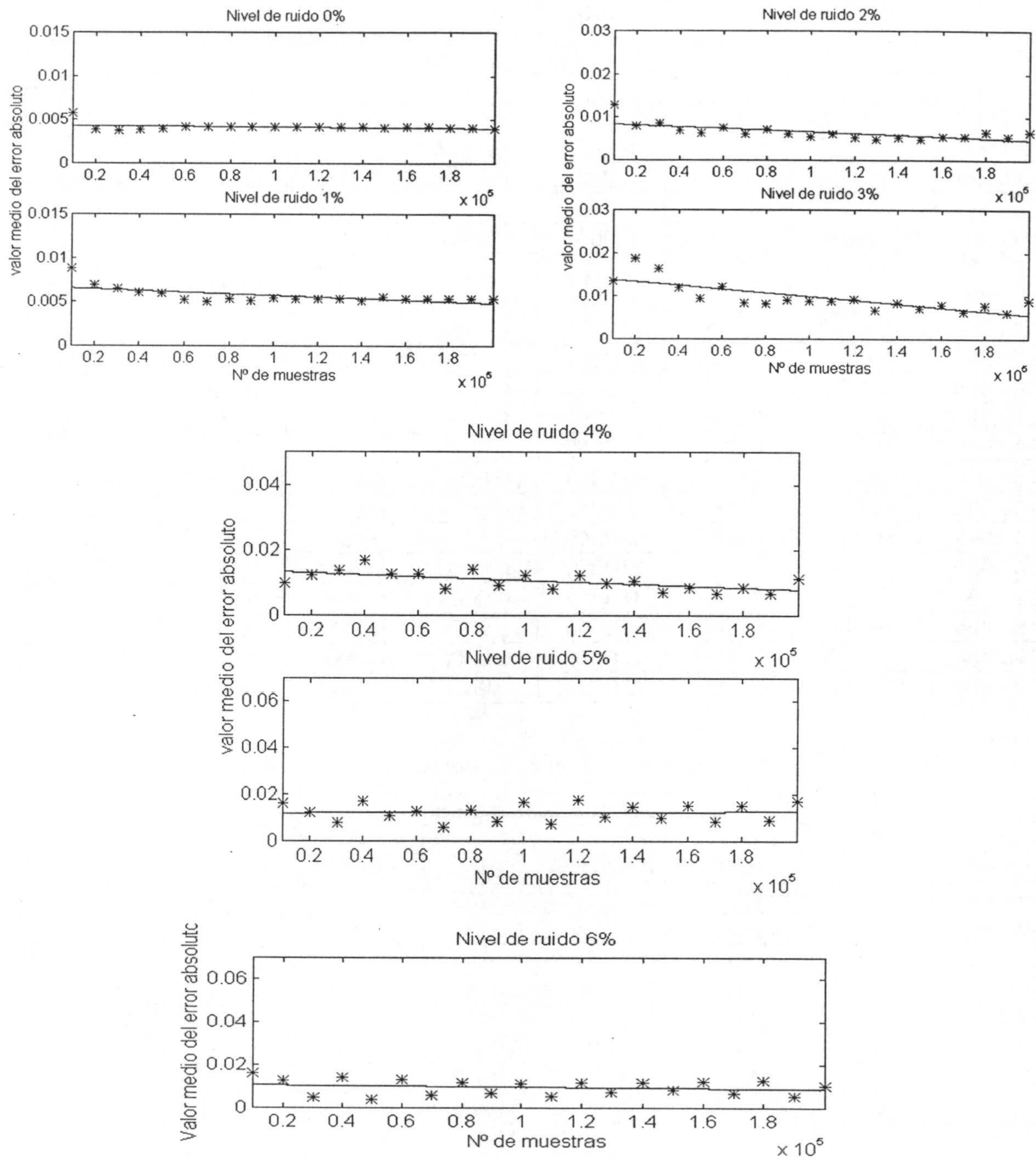


Figura A3.8.1. Valor medio del error absoluto frente al número de muestras para distintos niveles de ruido.

Tabla A3.8. 3. Ecuaciones de las rectas de ajuste correspondientes a la Figura A3.8.1, valor del error absoluto para 100.000 muestras y n° de muestras para que el error absoluto sea 0.02.

Nivel de error (%)	Ecuación de la recta	ϵ para $m=100.000$	m_L para $\epsilon = 0.02$
0	$\epsilon = - 0.17e-8 m + 0.004$	0.0042	<10.000
1	$\epsilon = - 0.92e-8 m + 0.007$	0.0057	<10.000
2	$\epsilon = - 1.9e-8 m + 0.009$	0.0066	<10.000
3	$\epsilon = - 4.3e-8 m + 0.014$	0.0099	<10.000
4	$\epsilon = - 2.9e-8 m + 0.013$	0.0106	<10.000
5	$\epsilon = 0.32e-8 m + 0.012$	0.0122	<10.000
6	$\epsilon = - 1.5e-8 m + 0.011$	0.0097	<10.000

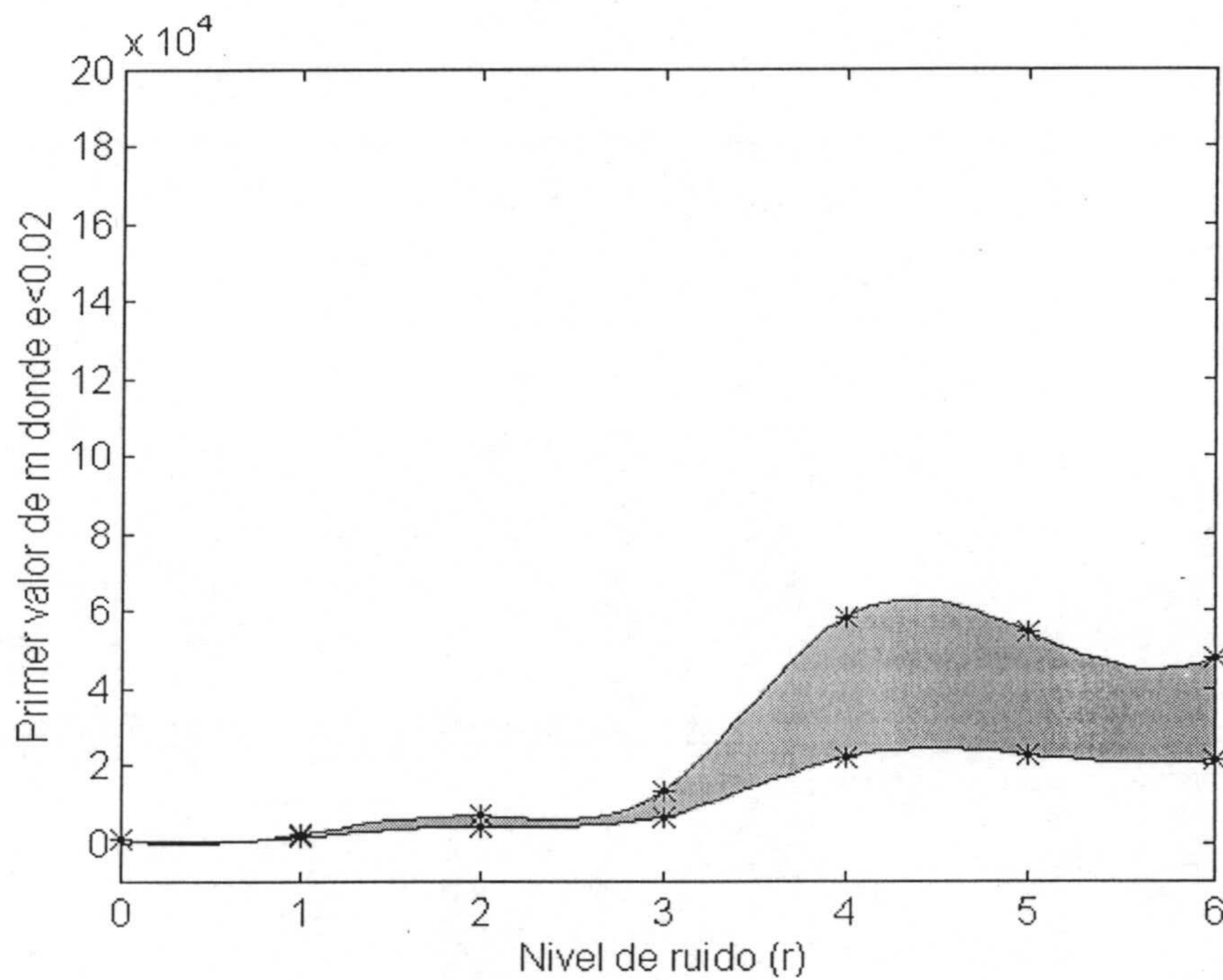
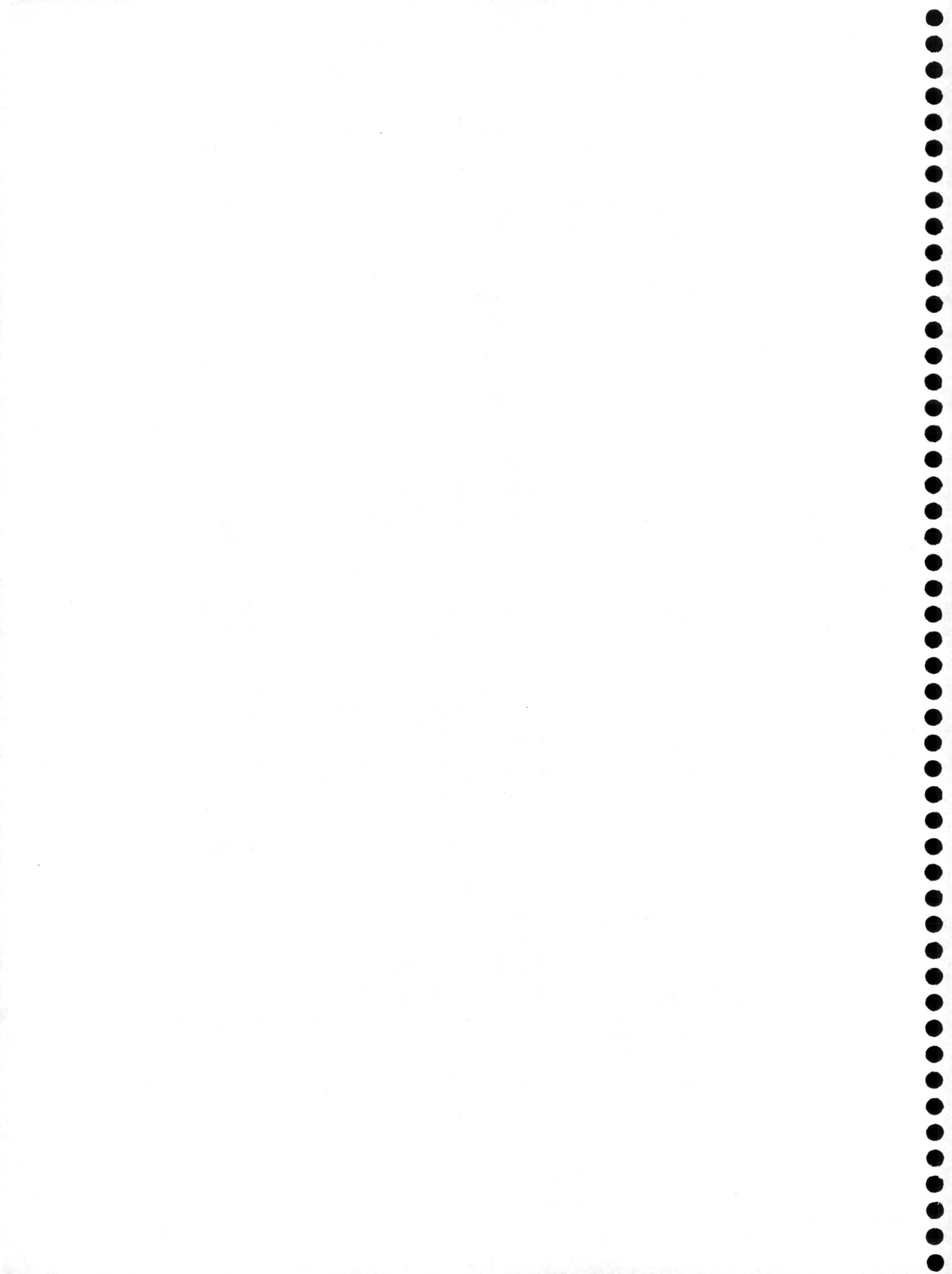


Figura A3.8.2. Primer valor de M donde el error absoluto es menor de 0.02.

Tabla A3.8.4. Primer valor de m con un error absoluto de 0.02, con la matriz A_8 para los distintos niveles de ruido.

Matriz A_8	Nivel de ruido uniforme (%)						
	0	1	2	3	4	5	6
Valor medio	1000	1600	4400	6800	22100	22700	21600
Desviación estándar	0	843	2547	6512	36186	31816	26412



REFERENCIAS BIBLIOGRÁFICAS

- [ALM99] L.B.Almeida, G.C.Marques, "*Nonlinear Blind Source Separation by Pattern Repulsion*". Lecture Notes in Computer Science, vol. 1607, pp.674-682, Springer-Verlag, 1999.
- [AMA85] S. Amari, "*Differential-geometrical methods in statistics*". New York: Springer-Verlag, 1985.
- [AMA95] S. Amari, A. Cichocki, H. H. Yang. "*Recurrent Neural Networks for Blind Separation*". Proceedings of 1995 International Symposium on Non-linear Theory and Applications (NOLTA'95), Las Vegas (USA), Vol. 1, pp. 37-42, December 1995.
- [AMA96] S. Amari, A. Cichocki, and H.H. Yang, "*A new learning algorithm for blind signal separation*". Advances in Neural Information Processing Systems, Vol.8, MIT Press: Cambridge, MA, pp. 757-763, 1996.
- [AMA97] S. Amari, J. F. Cardoso, "*Blind source separation – semiparametric statistical approach*". IEEE Trans. on Signal Processing, vol. 45, pp. 2692-2700, 1997.
- [AMA98a] S. Amari, "*Natural gradient works efficiently in learning*". Neural Computation, vol. 10, pp. 251-175. 1998.
- [AMA98b] S. Amari, A. Cichocki, "*Adaptive blind signal processing – Neural networks approach*". Proceedings of the IEEE vol. 86, nº 10, pp. 2026-2048, 1998.
- [BAC98] A. D. Back, A. S. Weigend, "*A first application of independent component analysis to extracting structure from stock returns*". International Journal of Neural Systems, vol. 9. Special Issue on Data Mining in Finance, 1998.
- [BAYxx] J. Bayliss, J. A. Gualtieri, R. F. Crompt, "*Analyzing hyperspectral data with independent component analysis*". xx
- [BEL95a] A. Bell, T. Sejnowski, "*An information-maximisation approach to blind separation and blind deconvolution*". Neural Computation, vol. 7, pp. 1129-1159, 1995.
- [BEL95b] A.J. Bell, T.J. Sejnowski, "*Fast blind separation based on information theory*". Proceedings of International Symposium on non-linear theory and applications, vol. 1, pp. 43-47, NTA Research Society of IEICE, Tokyo (Japan), 1995.
- [BEL97] A.J. Bell, T.J. Sejnowski, "*The independent components of natural scenes are edge filters*". Vision Research, Vol.37, No. 23, pp.3327-3338, 1997.

- [BOZ96] S. Bozinovski, H-L. Nguyen Thi, "*Séparation de sources à bande large dans un mélange convolutif*". Proc. of Ecole des Techniques Avancées en Signal Image Parole, pp. 303-310, Sept. 2-6, Grenoble (France), 1996.
- [BUR98] C.S.Burrus, J.H.McClellan, A.V.Oppenheim, T.W.Parks, R.W.Schafer, H.W.Schuessler, "*Ejercicios de tratamiento de la señal*", Prentice Hall, 1998.
- [CAP96] V. Capdeville, C. Servière, "*Séparation de signaux de machines tournantes*", Proceedings of Ecole des Techniques Avancées en Signal Image Parole, pp. 311-318, Grenoble, France, Sept. 2-6, 1996.
- [CAR86] A. B. Carlson, "*Communication Systems*", 3rd. Edit. McGraw-Hill, 1986.
- [CAR89] J. F. Cardoso, "*Source separation using higher order moments*". ICASSP pp. 2109-2112, Glasgow, 1989.
- [CAR93] JF. Cardoso, A. Souseloumia, "*Blind beamforming for non-Gaussian signals*". IEEE proceedings (London), Part F, vol. 140, pp.362-370, 1993.
- [CAR96a] J.F. Cardoso, "*Entropic contrasts for source separation*", NIPS'96 Workshop on blind signal processing, organized by A. Cichocki at Snomass, Colo. To appear as a chapter in the book UNSUPERVISED ADAPTIVE FILTERING. S. Haykin ed. New York: Wiley.
- [CAR96b] J.F. Cardoso, B. Laheld, "*Equivariant adaptive source separation*". IEEE Trans. on signal processing, vol. 44, n° 12, pp. 3017-3031, 1996.
- [CAR97] J. F. Cardoso, "*Statistical principles of source separation*". Proceedings of the SYSID'97, 11th IFAC Symposium on System Identification, pp. 1837-1844, 1997.
- [CAR98] J. F. Cardoso, "*Blind signal separation: statistical principles*". Proceedings of the IEEE, vol. 86, n° 10, pp.2009-2025, 1998.
- [CAR99] J. F. Cardoso, "*High order contrasts for independent component analysis*". Neural Computation, vol. 11, pp. 157-192, 1999.
- [CHA96] E, Chaumette, D. Muller, "*Séparation de sources aux ordres supérieurs: application au garbling en radar secondaire*", Ecole des Techniques Avancées en Signal Image Parole, pp. 245-253, Grenoble, France, Sept. 2-6, 1996.
- [CHE96] P. Chevalier, "*Les statistiques d'ordre supérieur en traitement d'antenne*", Proceedings of Ecole des Techniques Avancées en Signal Image Parole, pp. 237-243, Grenoble, France, Sept. 2-6, 1996.

- [CHE99] P. Chevalier, V. Capdeville, P. Comon, "Performance of HO blind source separation methods: experimental results on ionospheric HF links". Proc of ICA'99, pp. 443-448, January 11-15, Aussois (France), 1999.
- [CHI93] C. Y. Chi, J. Y. Kung, "A phase determination method of non minimum phase ARMA system by a single cumulant sample". IEEE trans. on SP, vol. 41, n°2, pp. 981-985, Feb. 1993.
- [CIC94] A. Cichocki, R. Unbehauen, E. Rummert, "Robust Learning Algorithm for Blind Separation of Signals". Electronic Letters, vol. 30, n° 17, pp. 1386-1387, 1994.
- [CIC95] A. Cichocki, W. Kasprzak, S.I. Amari, "Multi-layer neural networks with a local adaptive learning rule for blind separation of source signals". Proceedings of NOLTA'95, Las Vegas (USA), Dec. 10- 14, pp. 61-65.
- [CIC96] A. Cichocki, R. Unbehauen, "Robust neural networks with on-line learning for blind identification and blind separation". IEEE Transactions on Circuits and Systems I, vol. 43, n° 11, pp.894-906, November 1996.
- [COH92] M. H. Cohen, A. G. Andreou, "Current-mode subthreshold MOS implementation of the Héroult-Jutten autoadaptive network". IEEE journal of solid-state circuits, vol. 27, n° 5, May 1992.
- [COM94] P. Comon, "Independent component analysis. A new concept?". *Signal Processing*, vol. 36, n°3, pp.287-314, April 1994.
- [COR89] T. Cormen, C.E. Leiserson, R.L. Rivest, "Introduction to algorithms". The MIT Press, 1989.
- [COV91] T. Cover, J. Thomas, "Elements of Information Theory". New York: Wiley, 1991.
- [D'UR96] G. D'Urso, "Les techniques de séparation de sources appliquées à la surveillance des installations EDF", Proceedings of Ecole des Techniques Avancées en Signal Image Parole, pp. 255-263, Grenoble, France, Sept. 2-6, 1996.
- [DEV95] Y. Deville, L. Andry. "Application of blind source separation techniques to multi-tag contactless identification systems". Proceedings of the Proceedings of 1995 International Symposium on Non-linear Theory and Applications (NOLTA'95), vol. 1, pp. 73-78. Las Vegas, USA, Dec. 10-14, 1995.
- [DEV96a] Y. Deville, L. Andry. "Application of blind source separation techniques to multi-tag contactless identification systems". IEICE Transactions on Fundamentals of Electronics, Communications and Computer Science, vol. E79-A, n° 10, pp. 1694-1699, Oct. 96.

- [DEV96b] Y. Deville, "Application of the Héroult-Jutten source separation neural network to multi-tag radio-frequency identification systems". Proceedings of Ecole des Techniques Avancées en Signal Image Parole, pp. 265-272, Grenoble, France, Sept. 2-6 1996.
- [DEV99] Y. Deville, J. Damour, N. Charkani, "Improved multi-tag radio-frequency identification systems based on new source separation neural networks". Proc of ICA'99, pp. 449-454, January 11-15, Aussois (France), 1999.
- [FEN99] M. Feng, K-D. Kammeyer, "Application of source separation algorithms for mobile communication environment". Proc of ICA'99, pp. 431-436, January 11-15, Aussois (France), 1999.
- [GAE90] M. Gaeta, J. L. Lacoume, "Source separation without a priori knowledge: the maximum likelihood solution", Proc. EUSIPCO, pp. 621-624, 1990.
- [GEL99] G. Gelle, M. Colas, G. Delaunay, "Separation of convolutive mixtures of harmonic signals with a temporal approach. Application to rotating machine monitoring". Proc of ICA'99, pp. 109-114, January 11-15, Aussois (France), 1999.
- [GER94] S. Van Gerven, D. Van Compernelle, H. L. Nguyen-Thi, C. Jutten. "Blind Separation of Sources: A comparative Study of a 2-nd and 4-th order solution". Signal Processing VII Theories and Applications 1994. Pp. 1153-1156.
- [GRA90] R. Gray, "Entropy and Information Theory". New York: Springer-Verlag, 1990.
- [GUP77] M. S. Gupta, "Electrical noise: fundamentals and sources", IEEE Press, 1977.
- [HAY99] S. Haykin, "Neural Networks", Prentice Hall, 1999.
- [HER85] J. Héroult, C. Jutten and B. Ans, "Détection de grandeurs primitives dans un message composite par une architecture de calcul neuromimétique en apprentissage non supervisé". Actes du Xeme colloque GRETSI, pp.1017-1022, Nice, France, 20-21, May, 1985.
- [HER86] J. Héroult, C. Jutten, "Space or time adaptive signal processing by neural network models". In Neural Networks for Computing: AIP Conference Proceedings, 151. New York. American Institute for Physics, 1986.
- [HER94] J. Héroult, C. Jutten, "Réseaux neuronaux et traitement du signal". Hermès, 1994.
- [HYV99] A. Hyvarinen, P. Hoyer, E. Oja, "Denoising of nongaussian data by independent component analysis and sparse coding". Proc of ICA'99, pp. 485-490, January 11-15, Aussois (France), 1999.

- [JAY57] E. Jaynes, "*Information theory and statistical mechanics*". Physical Review, vol. 106, pp. 620-630; "Information theory and statistical mechanics II". Physical review, vol. 108, pp. 171-190, 1957.
- [JAY82] E. Jaynes, "*On the rationale of maximum-entropy methods*". Proceedings of the IEEE, vol. 70, pp. 939-952, 1982.
- [JOH94a] R. A. Johnson. "*Probabilidad y estadística para ingenieros de Miller y Freund*". Prentice Hall, 1994.
- [JOH94b] N. L. Johnson, S. Kotz, N. Balakrishan, "*Continuous univariate distributions*", Vol. 1, 2^a Edition, John Wiley & Sons, 1994.
- [JOH95] N. L. Johnson, S. Kotz, N. Balakrishan, "*Continuous univariate distributions*", Vol. 2, 2^a Edition, John Wiley & Sons, 1995.
- [JUT88] C. Jutten, J. Héroult, "*Independent component analysis versus PCA*". Proc. EURASIP, Signal Processing IV, Elsevier, pp. 643-646, Grenoble (Francia), 1988.
- [JUT91] C. Jutten, and J. Héroult, "*Blind Separation of sources, part I, II, III .*". *Signal Processing*, vol.24, pp.1-29, 1991.
- [JUT97] C. Jutten, "*From source separation to independent component analysis*", Proceedings of the European Symposium on Artificial Neural Networks (ESANN'97), Bruges, April 16-18, 1997, pp. 243-248, D facta, Brussels, 1997.
- [KAR94] J. Karhunen, and J. Joutsensalo, "*Representation and separation of signals using non-linear PCA type learning*", *Neural Networks*, vol.7, n°1, pp.113-127, 1994.
- [KAR95a] J. Karhunen, E. Oja, L. Wang, R. Vigario, J. Joutsensalo, "*A class of neural networks for independent component analysis*". Proceedings IEEE International Conference On Neural Networks, pp. 995-1000, Perth, Australia November 1995.
- [KAR95b] J. Karhunen, E. Oja, L. Wang, R. Vigario, J. Joutsensalo, "*A class of neural networks for independent component analysis*". Lab. Of Computer And Information Science, Helsinki, Univ. Of Technology (Espoo, Finland). Technical Report A28, October 1995.
- [KAR95c] J. Karhunen, J. Joutsensolo, "*Generalization of principal component analysis, optimization problems and neural networks*". *Neural Networks* vol. 8, n°4, pp. 549-562, 1995.

- [KAR96] J. Karhunen, "Neural approaches to independent component analysis and source separation". Proc. 4th European Symposium on Artificial Neural Networks (ESANN'96), pp. 249-266, Bruges, April 1996.
- [KAR97] J. Karhunen, E. Oja, L. Wang, R. Vigario, J. Joutsensolo, "A class of neural networks for independent component analysis". IEEE Transactions on neural networks, vol. 8, no. 3, pp. 486-504, May 1997.
- [KOH90] T. Kohonen, "The self organizing map", *Proceedings of the IEEE*, vol.7, no. 9, pp. 1464-1480, Sept. 1990.
- [KUL68] S. Kullback, "Information theory and statistics". Gloucester, MA:Peter Smith, 1968.
- [KUN96] M. Kunt, "Traitement numérique des signaux". Presses polytechnique et Universitaires Romandes. 1996.
- [KOP99] I. Kopriva, A. Persin, "Blind separation of optical tracker responses into independent components discriminates optical sources". Proc of ICA'99, pp. 31-36, January 11-15, Aussois (France), 1999.
- [NUZ99] D. Nuzillard, J-M. Nuzillard, "Blind source separation applied to non-orthogonal signals". Proc of ICA'99, pp. 25-30, January 11-15, Aussois (France), 1999.
- [LAC88] J. L. Lacoume, P. Ruiz, "Sources identification: a solution based on the cumulants". In Proc. 4th ASSP Workshop spectral estimation modelling, pp. 199-203, Minneapolis, 1988.
- [LAC92] J. L. Lacoume, P. Ruiz, "Séparation of independent sources from correlated inputs". IEEE on trans. SP, vol. 40, n° 12, pp. 3074-3078, Dec. 1992.
- [LAC97] J.L. Lacoume, P.O. Amblard, P. Comon, "Statistiques d'ordre supérieur pour le traitement du signal", Masson 1997.
- [LAH94] B. Lahed, "Séparation auto-adaptative de sources. Implantation et performances". Tesis, Escuela nacional superior de telecomunicaciones de París, 1994.
- [LEE98] T. Lee, "Independent component analysis". Kluwer Academic Publishers, 1998.
- [MAN95] A. Mansour, C. Jutten. "Fourth-Order criteria for blind separation". IEEE Transactions on signal processing, Vol. 43 n°8, August 1995, pp. 2022-2025.
- [MAR95] H. Marsman, "A neural net approach to the source separation problem". Tesis, Universidad de Twente (Nederlands), 1995.

- [MAR97] R. Martín Clemente, J. I. Acha. "Blind separation of sources using a new polynomial equation". *Electronics letters*, vol. 33, n°1, pp.176-177, 1997.
- [MAR99] R. Martín Clemente, J. I. Acha. "A new algorithm for the adaptive separation of sources". Proc of ICA'99, pp. 257-260, January 11-15, Aussois (France), 1999.
- [MAT95] K. Matsuoka, M. Ohya, M. Kawamoto, "A neural net for blind separation of nonstationary signals". *Neural Networks*, vol. 8, n° 3, pp. 411-419, 1995.
- [MEN95] W. Mendenhall, T. Sincich. "Probabilidad y estadística para ingeniería y ciencias". Prentice Hall, 1995.
- [MIT93] S. K. Mitra, J. F. Kaiser (Eds.). "Handbook for digital signal processing". John Wiley & Sons, 1993.
- [NAD97] J. P. Nadal, N. Parga, "ICA: conditions on cumulants and information theoretic approach". Proc. 5th European Symposium on Artificial Neural Networks (ESSAN'97), pp. 281-290, Bruges, 16-18 April, 1997.
- [NGO99] J.T. Ngo, N. A. Bhadkamkar, "Adaptive blind separation of audio sources by a physically compact device using second-order statistics". Proc of ICA'99, pp. 257-260, January 11-15, Aussois (France), 1999.
- [NGU95] H. L. Nguyen, C. Jutten, "Blind source separation for convolutive mixtures". *Signal Processing*, vol. 45, pp. 209-229, 1995.
- [OJA85] E. Oja, J. Karhunen, "An stochastic approximation of the eigenvectors and eigenvalues of the expectation of a random matrix". *J. Math. Anal. Appl.* vol. 106, pp. 69-84, 1985.
- [OJA91] E. Oja, J. Ogawa, "Learning in non-linear constrained Hebbian networks", in T. Kohonen, K. Makisara, O. Simula, J. Kangas (Eds.). *Artificial Neural Networks*, Elsevier, Amsterdam, 1991, pp. 16-35.
- [OJA95] E. Oja, J. Karhunen, L. Wang, and R. Vigario, "Principal and independent components in neural-networks -recent developments", In *Proc. Italian Workshop on Neural Networks, WIRN'95*, Vietri, Italy, 1995.
- [OJA97] E. Oja, "The non-linear PCA learning rule in independent component analysis". *Neurocomputing* vol. 17, pp. 25-45, 1997.
- [PAP84] A. Papoulis, "Probability, random variables and stochastic processes". 2nd edition, New York: McGraw Hill, 1984.
- [PAR98] A. Paraschiv-Ionescu, c. Jutten, A. M. Ionescu, A. Chovet, A. Rusu, "High performance magnetic field smart sensor arrays with source separation", First international conference on modelling and simulation of microsystems (MSM98), Santa Clara (USA), April 6-8, 1998.

- [PEI91] A. Peinado, J.M. López, V. E. Sánchez, J. C. Segura, A. J. Rubio, "Improvements in HMM-based isolated word recognition systems". IEEE Proc. I, vol. 138, nº 3, pp. 201-206, June 1991.
- [PLU93] M. Plumbey, "A Hebbian/anti-Hebbian network which optimizes information capacity by orthonormalizing the principal subspace". Proc. IEEE conf. Artificial Neural Networks, pp. 86-90, Brighton, UK, May 1993.
- [PRI96] A. Prieto, "Una visión actual sobre la Neurocomputación", Servicio de Publicaciones de la Universidad de Granada, ISBN, 84-338-2229-2, 1996.
- [PRI97] A. Prieto, C. G. Puntonet, B. Prieto, and M. Rodríguez-Alvarez, "A Competitive Neural Network for Blind Separation of Sources Based on Geometric Properties". *Lecture Notes in Computer Science*, vol. 1240, pp.1095-1106, Springer-Verlag, 1997.
- [PRI98] A. Prieto, C. G. Puntonet, B. Prieto; "A Neural Learning Algorithm for blind separation of sources based on geometric properties". *Signal Processing*, Vol.64, No. 3, pp. 315-331, 1998.
- [PRI99a] A. Prieto, B. Prieto, C.G. Puntonet, A. Cañas, P. Martín-Smith, "Geometric separation of linear mixtures of sources: Application to speech signals", International workshop on Independent Component Analysis and Blind Signal Separation (ICA'99) , pp. 295-300, Aussois, France, January 11-15, 1999.
- [PRI99b] B. Prieto, A. Prieto, C. G. Puntonet, M. Rodríguez, "A software tool to simulate linear mixture separation algorithms", *Demostración en el "1st. Inter. Workshop on Independent Component Analysis and Signal separation (ICA'99)", January 11-15, 1999 Aussois, France.*
- [PRO97] J.G. Proakis, D.G. Manolakis, "Tratamiento de señales", tercera edición. Prentice Hall, 1997.
- [PUG99] A. T. Puga, A. P. Alves, "An experiment on comparing PCA and ICA in classical transform image coding", proceedings of ICA'99, pp. 105-108, Aussois, France, Jan 11-15, 1999.
- [PUN95a] C. G. Puntonet, A. Prieto, C. Jutten, M. Rodríguez-Alvarez, and J. Ortega, "Separation of sources: a geometry-based procedure for reconstruction of n -valued signals". *Signal Processing*, vol. 46, no. 3, pp. 267-284, 1995.
- [PUN95b] C.G. Puntonet, M. Rodríguez-Alvarez,; A. Prieto, "A Geometrical Based Procedure for Source Separation Mapped to a neural Network", *Lecture Notes in Computer Science*, Vol.930, pp.736-743, Springer Verlag, 1995.

- [PUN95c] C.G.Puntonet, A. Prieto, "*An adaptive geometrical procedure for blind separation of sources*", Neural Processing Letters, Vol.2, No.5, pp.23-27, Sept. 1995.
- [PUN95d] C. G. Puntonet, A. Mansour, C. Jutten, "*Geometrical Algorithm for Blind Separation of Sources*", Proc. of 15me. Colloque sur le Traitement du Signal et des Images (GRETSI'95), Vol.1, pp. 273-276. Juan les Pins, 18-22 Sept. 1995.
- [PUN96a] C.G.Puntonet, M. Rodriguez, A.Prieto. "*Adaptive blind separation of signals through a new geometrical approach*". IASTED-96 (Artificial Intelligence, Expert Systems and Neural Networks, Honolulu, Hawaii (USA), 19-22 Agosto, 1996.
- [PUN96b] C.G.Puntonet, A.Prieto, J. Ortega. "*New geometrical approach for blind separation of sources mapped to a neural network*". NICROSP-96, International Workshop on Neural Networks for Identification, Control, Robotics, and Signal-Image Processing, pp.174-182, 21-23 Agosto, 1996, Venecia, Italia.
- [PUN96c] C.G.Puntonet, M. Rodriguez, A.Prieto. "*Blind separation of signals through a geometrical approach*". IIZUKA-96, 4th International Conference on Soft Computing, pp.667-670, 30 Septiembre al 5 Octubre, 1996, Fukuoka, Japon.
- [PUN97a] C. G. Puntonet, A. Prieto, "*Geometric approach for blind separation of signals*", *Electronic Letters*, vol.33, no. 10, pp. 835-836, 1997.
- [PUN97b] Puntonet, C.G., Rodriguez-Alvarez, M, Prieto, A: "*A geometric method for blind separation of sources*". Artificial Intelligence and Soft Computing, IASTED-97, Banff, Canada, 27-31 July, pp. 372-375, IASTED Acta Press, 1997.
- [PUN98a] C.G.Puntonet, A. Prieto, "*Neural net approach for blind separation of sources based on geometric properties*", Neurocomputing Vol. 18, No.1-3, pp. 141-164, Jan. 1998.
- [PUN98b] C.G. Puntonet, M. R. Alvarez, A. Prieto ,B. Prieto, "*Separation of Sources in a Class of Post-Nonlinear Mixtures*", European Symposium on Artificial Neural Networks (ESANN'98), pp. 321-326, IEEE, Bruges- April 22-23-24, 1998.
- [PUN99] C.G. Puntonet; M. R. Alvarez; A. Prieto; B. Prieto. "*Separation of Speech Signals for Nonlinear Mixtures*". Lecture Notes in Computer Science, vol. 1607, pp.665-673, Springer-Verlag, 1999.
- [RAB78] L.R. Rabiner, and R.W.Schafer, "*Digital Processing Signals*", Prentice-Hall, 1978.

- [RIS99] T. Ristaniemi, J. Joutsensalo, "On the performance of blind source separation in CDMA downlink". Proc of ICA'99, pp. 437-442, January 11-15, Aussois (France), 1999.
- [ROD96] V. Rodellar, V. García, P. Gómez, E. Martínez, V. Peinado. "Optimal arithmetic dimension of adaptive filters for speech characterization". Proc. Of the MELECON'96, Bari, Italy, 1996.
- [RUM86] D.E. Rumelhart, D. Zipser, "Feature discovery by competitive learning". In: Parallel Distributed processing, Rumelhart, D.E. & McClelland, J.L., MIT Press; vol. I, pp. 151-193, 1986.
- [SAH96] H. Sahlin, H. Broman, "Blind separation of images", Proceedings of the 30th ASILOMAR conference on signals, systems and computers, 1996.
- [SAN89] T. Sanger, "Optimal unsupervised learning in a single-layer linear feedforward neural network". Neural Networks, vol. 12, pp. 459-473, 1989.
- [SCH99] I. Schief, M. Stetter, J. E. M. Mayhew, S. Askew, N. McLoughlin, J. B. Levitt, J. S. Lund, K. Obermayer, "Blind separation of spatial signal patterns from optical imaging records". Proc of ICA'99, pp. 179-184, January 11-15, Aussois (France), 1999.
- [SCO96] M. Scott, T. Jung, A. Bell, D. Ghahremani, T. Sejnowski, "Blind source separation of event-related brain response components". Society for psychophysiological research, Oct. 1996.
- [SED88] R. Sedgewick, "Algorithms". Second edition, Addison-Wesley publishing company, 1988.
- [SER96] P. Serra, J. Parramon, E. Valderrama, J. L. Huertas, "Circuitaria para el registro de señal en interfases neurales de tipo regenerativo". II Workshop de Iberchip, Sao Paulo, Brasil, 1996.
- [SHO80] J. Shore, R. Johnson, "Axiomatic derivation of the principle of maximum entropy and the principle of minimum cross-entropy". IEEE Transaction on information theory, vol. IT-26, pp. 26-37, 1980.
- [SIL91] F. Silva, L. Almeida, "A distributed solution for data orthonormalization". in Artificial Neural Networks Proc. ICANN-91, pp. 943-948, T. Kohonen et al., Eds., Amsterdam, The Netherlands: North-Holland, 1991.
- [SIL97] H. F. Silverman, W. R. Patterson, J. L. Flanagan, D. V. Rabinkin, "A digital processing system for source location and sound capture by large microphone arrays". Proc. of ICASSIP'97, Munich (Alemania), 1997.
- [STU94] A. Stuart, J. K. Ord, "Kendall's advanced theory of statistics", Vol. 1: Distribution theory. 6^a Edición. Edward Arnold, 1994.

- [TAL97] A. Taleb, C. Jutten, "Non-linear source separation: the post-non-linear mixtures", ESANN'97, pp. 279-299, 16-18 April, Bruges, 1997.
- [TAL98] A. Taleb, C. Jutten, S. Olympief, "Source separation in post non-linear mixtures: an entropy-based algorithm", ICASSP'98, Seattle (USA).
- [TAL99] A. Taleb, J. Solé, C. Jutten, "Blind Inversion of Wiener Systems", Lecture Notes in Computer Science, vol. 1607, pp. 655-664, Springer-Verlag, 1999.
- [THI95] H. N. Thi, C. Jutten, "Blind source separation for convolutive mixtures". Signal Processing, vol. 45, n° 2, pp. 209-299, 1995.
- [THI96] N. Thirion, J. Mars, F. Glangeaud, "Séparation d'ondes dans le cas des signaux de prospection sismique", Proceedings of Ecole des Techniques Avancées en Signal Image Parole, pp. 357-366, Grenoble, France, Sept. 2-6, 1996.
- [TOR99] K. Torkkola, "Blind separation for audio signals – are we there yet?". Proc of ICA'99, pp. 239-244, January 11-15, Aussois (France), 1999.
- [VIG99] R. Vigario, J. Sarela, V. Jousmaki, E. Oja, "Independent component analysis in decomposition of auditory and somatosensory evoked fields". Proc of ICA'99, pp. 167-172, January 11-15, Aussois (France), 1999.
- [VET99] R. Vetter, J. M. Vesin, P. Celka, U. Scherrer, "Observer of the autonomic cardiac outflow in humans using non-casual blind source separation". Proc of ICA'99, pp. 161-166, January 11-15, Aussois (France), 1999.
- [VIT89] E. A. Vittoz, X. Arreguit, "CMOS integration of Héroult-Jutten cells for separation of sources". Workshop on analogue VLSI and neural systems, Portland (Oregon), May 8, 1989.
- [WES99] A. Westner, V. M. Bove, "Blind separation of real world audio signals using overdetermined mixtures". Proc of ICA'99, pp. 251-256, January 11-15, Aussois (France), 1999.
- [WU99] H-C. Wu, J. C. Principe, "Simultaneous diagonalization in the frequency domain (SDIF) for source separation". Proc of ICA'99, pp. 245-250, January 11-15, Aussois (France), 1999.
- [YAN97] H. H. Yang, and S. Amari, "Adaptive on-line learning algorithms for blind separation maximum entropy and minimum mutual information", Neural Computation, vol. 9, pp. 1457-1482, 1997.
- [YPM99] A. Ypma, P. Pajunen, "Rotating machine vibration analysis with second-order independent component analysis". Proc of ICA'99, pp. 37-42, January 11-15, Aussois (France), 1999.
- [ZHU95] J. Zhu, X.R. Cao, R.W. Liu, "Blind source separation based on output independence – theory and implementation". Proceedings of 1995

International Symposium on Non-linear Theory and Applications
(NOLTA'95), vol.1, pp. 97-102, Las Vegas (USA), December 1995.

Biblioteca Universitaria de Granada



01066731