TESIS DOCTORAL

DESARROLLO DE TÉCNICAS DE CODIFICACIÓN DE AUDIO BASADAS EN MODELOS DE SEÑAL PARAMÉTRICOS

Pedro Vera Candeas
email: pvera@ujaen.es

Directores:
Nicolás Ruiz Reyes, Manuel Rosa Zurera

Departamento de Teoría de la Señal y Comunicaciones
Escuela Politécnica Superior
UNIVERSIDAD DE ALCALÁ
Septiembre 2006

Esta tesis doctoral no hubiera sido posible sin el apoyo de mi familia y amigos. Gracias a todos y, en especial, a mis padres y a Ana Lucía.

Prólogo

Esta tesis doctoral supone la continuación de una línea de investigación iniciada a principios de los 90 por el co-Director de esta tesis doctoral, el Dr. D. Manuel Rosa Zurera, dentro del ámbito de la compresión de la señal de audio utilizando, en su caso, descomposiciones basadas en la transformada wavelet. Posteriormente, esta línea de investigación fue continuada por el otro co-Director de esta tesis, el Dr. D. Nicolás Ruiz Reyes, ampliando la descomposición al uso de transformada wavelet packets adaptada a la señal. Adicionalmente, en el mismo grupo, se ha desarrollado también una tesis en compresión de audio, aunque esta vez minimizando el retardo del sistema, por el Dr. D. Damián Martínez Muñoz, también co-dirigida por el Dr. D. Manuel Rosa Zurera.

Como resultado de esta labor investigadora se ha adquirido por este grupo un profundo conocimiento de la señal de audio y de las posibilidades tecnológicas en el campo de su compresión. Fruto de este conocimiento surgieron nuevas líneas investigación, destacando la investigación en el campo de los modelos de señal adaptativos y su aplicación a la compresión, modificación y síntesis de señales de audio, que han permitido el desarrollo de esta tesis.

Esta línea de trabajo, en el ámbito de la codificación de señal, no está aún agotada. Así, si bien con las herramientas propuestas no parece posible nuevas contribuciones importantes en compresión de audio, si es posible el empleo de estas técnicas en otros problemas. En el campo del audio, un problema a resolver es la transmisión de audio a través de Internet (*Internet audio streaming*) a régimen binario bajo y adaptativo a las condiciones cambiantes de la red. Otro tema de interés es el empleo de las herramientas de señal desarrolladas para analizar otro tipo de señales, como la señal de electrocardiograma o la señal ultrasónica, donde ya se han realizado algunos avances.

La realización de esta tesis doctoral me ha permitido iniciarme en el mundo de la investigación, pudiendo así realizar de forma completa las funciones de un Profesor de Universidad. Además, me satisface personalmente participar como investigador en los inicios del grupo de investigación *Tratamiento de Señales en Sistemas de Telecomunicación*, formado por personas que han realizado el doctorado en los últimos años o están en proceso de realización, todas ellas pertenecientes al Departamento de Ingeniería Electrónica, de Telecomunicación y Automática de la Universidad de Jaén. En este grupo tenemos puestas muchas esperanzas en el desarrollo de una investigación de calidad dentro de la Universidad de Jaén y en el ámbito de las tecnologías de la información y las comunicaciones.

Quiero hacer constar mi agradecimiento al Dr. D. Francisco López Ferreras, Director del Grupo de Señales y Circuitos del Departamento de Teoría de la Señal y Comunicaciones de la Universidad de Alcalá, por haber hecho posible la realización de esta tesis cuando en la

Universidad de Jaén no existían las condiciones necesarias. Además, este agradecimiento se hace extensivo al resto de componentes de dicho grupo por la ayuda prestada y las facilidades obtenidas siempre que se les ha requerido.

Merece una mención especial el co-Director de esta tesis doctoral, Dr. D. Nicolás Ruiz Reyes, por su entusiasmo y dedicación. El contraste con sus ideas ha sido fundamental para la culminación del trabajo de investigación reflejado en esta tesis doctoral. Han sido muchas las horas de trabajo que han sido necesarias para que los frutos de esta tesis salgan a la luz.

Finalmente, el agradecimiento al Departamento de Ingeniería Electrónica, de Telecomunicación y Automática de la Universidad de Jaén, al que pertenezco, por la facilidades prestadas para la realización de esta labor, y especialmente, a mis compañeros de departamento en la Escuela Politécnica Superior de Linares, por su participación en la ardua tarea de la realización de las pruebas de audición. No quiero olvidarme de mis compañeros de área de conocimiento, Pepe, Raúl, Damián, Juan Pedro, Fernando, Pedro y José Miguel, por la ayuda y confianza que siempre me han ofrecido.

Resumen

Conforme los sucesivos estándares de codificación de audio ISO/MPEG, basados en codificación de forma de onda y por transformada, han ido avanzando, se ha llegado al límite de esta tecnología en relación a la reducción del régimen binario. Por esta causa, han ido surgiendo nuevos avances en relación a la modelización de la señal que permiten, con unos pocos parámetros, codificar componentes de la señal de audio. En este sentido es de uso común, actualmente, utilizar MP3-pro que se basa en la replicación de bandas espectrales de alta frecuencia en función de ciertos parámetros y en la codificación de forma de onda de las bajas frecuencias.

El uso de modelos de señal paramétricos ha sido utilizado desde los años 90 como una herramienta de procesamiento de señales musicales. Esta tecnología se ha ido trasladando al campo del audio, al principio en codificadores mixtos basados en transformada que, en algunas circunstancias, se aprovechan de diferentes modelos para parametrizar las partes tonales o ruidosas de la señal. Posteriormente, han ido surgiendo nuevas propuestas que utilizan al máximo estos modelos, dividiendo la señal de audio en tonos, transitorios y ruido, para parametrizar por completo la señal.

Esta tesis se centra en la consecución de un codificador completamente paramétrico de audio que, en ningún momento, codifique la forma de onda de la señal. Para lograr este fin se han realizado avances en el estado del arte en relación al modelo sinusoidal, de transitorios y de ruido.

Respecto al modelo sinusoidal se incluye en esta tesis un algoritmo iterativo basado en *matching pursuits* que extrae el tono perceptualmente más importante en cada iteración. Además, el proceso se detiene cuando se han extraído todos los tonos perceptualmente importantes de la señal. Incluso se cuantifican las amplitudes de los tonos de forma transparente, con un número de bits variable usando principios psicoaústicos y sin enviar información lateral.

Para el modelo de transitorios se ha hecho un esfuerzo significativo con el fin de desarrollar un modelo paramétrico de baja complejidad que permita una adecuada caracterización de esta componente de la señal. En este sentido, se ha definido un modelo iterativo basado en *matching pursuits* con un diccionario de funciones wavelet packets. También se ha introducido un modelo de mayor complejidad, y con mejores resultados, que incluye en el diccionario tanto funciones wavelet packets como exponenciales complejas.

El residuo de los anteriores modelos se caracteriza típicamente como ruido, parametrizando su energía en tiempo y frecuencia. Para este modelo se ha hecho una revisión de las herramientas disponibles, habiendo utilizado un predictor lineal en frecuencia modificada logarítmicamente (adaptándose a las características del oído humano) para expresar la energía en frecuencia.

Con todas estas herramientas se ha estructurado un codificador de audio completamente

paramétrico. Se incluye en el funcionamiento del codificador un algoritmo de segmentación adaptativa del eje temporal muy flexible, así como los procesos de cuantificación de parámetros necesarios teniendo en cuenta siempre criterios perceptuales.

Los estudios teóricos y los desarrollos efectuados han dado lugar a un codificador de alta calidad de señales CD-audio que emplea una media 16 kbits/s (0,36 bits/muestra por canal), resultando una opción ventajosa a bajo régimen binario con respecto al estándar AAC actualmente establecido en el mercado.

Abstract

The bit rate reduction boundary of transform based coders, which quantize the waveform of the signal, has been almost reached by the last ISO/MPEG standards. As a consequence, a novel research domain has appeared in order to reduce the audio bit rate: parametric audio modelling. For example, the MP3-pro standard extracts the high frequency bands from both the waveform coded low frequency bands and some parameters, this process is known as spectral band replication.

Parametric models for musical signals have been utilized since the nineties. Nowadays, these tools are being applied to audio coding. Firstly, these models were included into mixed coders, which were basically waveform coders, but they sometimes made use of sinusoidal or noise models to lower the bit rate. Later, fully parametric audio coders, which decompose the audio signal into sinusoids, transients and noise, have been proposed.

The implementation of a fully parametric audio coder is the main objective of this thesis. Therefore, new advances, in regard to sinusoidal, transient and noise modelling, have been accomplished for achieving high quality and low bit rate audio coding.

In relation to sinusoidal modelling we propose a perceptual matching pursuits algorithm which extracts the most perceptually meaningful tone at each iteration. Also, a perceptual stopping criterion is presented: the algorithm is halted when all the psychoacoustic meaningful tones are extracted. Besides, tone amplitudes are quantized in a variable number of bits achieving transparent quantization without sending additional side information.

Transient modelling has been advanced because we have made an effort to develop a low complexity parametric model that is adapted to different transient signals. As a result, we propose a matching pursuits algorithm with a wavelet packets dictionary and a fast procedure to update correlations. Also, a more complex model but with better results is treated, this model is based on matching pursuits algorithm with mixed (wavelet packets & complex exponentials) dictionary.

The remaining of the previous models is analized as a noise signal, extracting its time and frequency energy characteristics. We have revised the techniques used in the literature and, finally, we have included a warped linear predictor in order to modelize the noise energy in frequency.

We define a fully parametric audio coder by using all these mentioned tools and by adding an adaptive segmentation algorithm (which has to be very flexible) and psychoacoustical information to quantize all the derived parameters.

These theoretical studies and accomplished developments have led to a high quality audio coder for CD-audio signals that uses an average of 16 kbits/s (0.36 bits/sample per channel). This coder can be a profitable alternative to the AAC standard currently established in the market.

Índice general

Índice de figuras

Índice de cuadros

Parte I

Planteamiento de la Investigación y Revisión de Conocimientos

Capítulo 1

Introducción

1.1. Contexto y localización de la investigación

La representación digital de señales de audio encontró en los años ochenta un estándar con la aparición de la tecnología del disco compacto (CD). Inevitablemente, todos los esquemas de codificación de señales de audio que han surgido desde entonces han tratado de comparar su calidad con la calidad CD. Esta se caracteriza por el uso de una frecuencia de muestreo de 44,1 kHz, para señales de audio cuyo ancho de banda es del orden de 20 kHz, siendo cada muestra PCM codificada con 16 bits.

Para transmitir las señales digitales resultantes, se necesitaría una velocidad de transmisión de 705,6 kbits/s por canal de audio, justificándose la necesidad de investigación para encontrar técnicas de codificación alternativas que permitan reducir el régimen binario manteniendo la calidad perceptual de la señal de audio decodificada.

La investigación en este campo no es nueva, apareciendo los primeros sistemas de codificación de audio digital con buena calidad y bajo régimen binario a finales de la década de los ochenta. Desde entonces, la investigación en esta línea se ha ido intensificando, debido fundamentalmente a las aportaciones realizadas por el grupo MPEG (*Moving Pictures Expert Group*), fruto de las cuales han surgido diversos estándares internacionales de codificación de audio. Estos estándares son los siguientes: MPEG-1 audio (ISO/IEC 11172-3, 1992), MPEG-2 audio (ISO/IEC 13818-3, 1994) y MPEG-4 (ISO/IEC 14496, 2000).

Sin embargo, a partir del año 2000 ha ido surgiendo la necesidad de reducir de manera significativa el régimen binario de la señal de audio con el objetivo de poder realizar transmisiones sobre Internet, así como incrementar el tiempo de señal que se puede almacenar en dispositivos portátiles de bajo coste. Para lograr este fin se ha apuntado la necesidad de cambiar la tecnología de codificación, pasando de utilizar transformadas tiempo-frecuencia para codificar la forma de onda de la señal, a desarrollar modelos de señal que extraigan parámetros de la señal de audio que son posteriormente codificados. El éxito de estos sistemas reside fundamentalmente en la gran compactación de energía al suponer un modelo de tres componentes: tonos, transitorios y ruido. Además, con este tipo de codificadores paramétricos es muy sencillo realizar modificaciones de la señal de audio a partir de los parámetros y suponen, adicionalmente, una herramienta prometedora para desarrollar esquemas de reconocimiento y separación de fuentes.

En esta tesis se aborda la revisión y desarrollo de modelos de señal como herramienta fun-

damental en el análisis de las señales de audio para sistemas de codificación paramétrica. Se ha conseguido incluir el uso de información psico-acústica en la extracción tonal del modelo sinusoidal, así como una representación adecuada de los transitorios y ruido de la señal, para alcanzar una reducción significativa del régimen binario manteniendo una buena calidad perceptual.

1.2. Justificación y objetivos de la investigación

Todo avance tecnológico se fundamenta en varias etapas: investigación básica, investigación aplicada, desarrollo y producción. En esta tesis doctoral se presenta un trabajo que combina las principales características de la investigación básica y de la investigación aplicada. Por un lado, se desarrollan nuevas estrategias de modelización de la señal de audio, que podrían ser útiles en muchas aplicaciones. Por otro, se propone un producto que, con ligeras modificaciones y mejoras, es susceptible de ser explotado.

La investigación llevada a cabo se fundamenta en la siguiente hipótesis de partida, siendo esta tesis doctoral el trabajo realizado para comprobar su veracidad:

HIPÓTESIS:

La completa parametrización de la señal de audio mediante el uso de modelos adaptativos, basados en la descomposición de la señal de audio en tonos, transitorios y ruido, y su codificación siguiendo criterios perceptuales proporciona una ganancia importante, en cuanto a régimen binario, comparada con la utilización de codificación de forma de onda con descomposiciones tiempo frecuencia.

Teniendo en cuenta el objetivo general de la investigación, es preciso plantear una serie de objetivos específicos, cuya consecución permita alcanzar el objetivo general:

- Definición de un algoritmo de segmentación adaptativa del eje temporal, para conseguir dividir la señal de audio en segmentos que podamos considerar casi estacionarios, a los cuales aplicar los modelos de señal paramétricos. De esta forma se pretende minimizar la distorsión de pre-eco.
- Realizar una extracción tonal basada en principios psicoacústicos que proporcione una herramienta capaz de extraer los tonos perceptualmente importantes de un segmento de audio.
- Desarrollar un modelo paramétrico de transitorios que se adapte a las características de la señal. El algoritmo debe ser lo suficientemente flexible como para poder parametrizar los diferentes tipos de transitorios que puedan aparecer en la señal de audio.
- Implementar un modelo de ruido que extraiga, de forma eficiente y con alta calidad, las características en tiempo y frecuencia de la señal residual de los modelos previos.
- Inclusión de algoritmos eficientes de codificación de los parámetros de cada modelo basados en criterios perceptuales.

El resultado final ha sido la propuesta de un codificador de audio basado en modelos de señal que consigue la completa parametrización de la señal de audio. Este codificador proporciona

regímenes binarios del orden de 16 kbits/s para todas las señales de prueba, manteniendo una alta calidad de la señal codificada.

1.3. Estructura de la tesis

En esta sección se presenta la estructura de la tesis que recoge el trabajo de investigación desarrollado. Se estructura en tres bloques temáticos. Cada bloque temático por su parte está compuesto por una serie de capítulos.

- **Planteamiento de la investigación y revisión de conocimientos.**

  Este bloque temático está compuesto de tres capítulos. El primero de ellos, que es en el que nos encontramos en este momento, se centra fundamentalmente en la presentación de los objetivos de la investigación y de la estructura de la tesis doctoral.

  En el segundo capítulo se presentan los fundamentos de los sistemas de codificación perceptual de audio y se realiza una revisión del estado del arte en relación a los sistemas de codificación de audio en general.

  El tercer capítulo está dedicado a la revisión de los conceptos más relevantes en relación a la codificación paramétrica de audio. Los aspectos más destacados que se tratan son el modelado sinusoidal, el modelado de transitorios y el modelado de ruido. Además, se incluyen los trabajos previos con mayor importancia en el uso de los anteriores modelos en codificadores de audio, ya sean totalmente paramétricos, o aquellos que, basados en transformada, incluyen alguna de estas herramientas.

  El cuarto capítulo se dedica al estudio de las descomposiciones atómicas. Los modelos de señal mediante los que se obtienen los parámetros de la señal de audio en esta tesis doctoral están basados en su mayor parte en descomposiciones atómicas. Se revisarán los diferentes métodos, tanto paralelos como iterativos, que existen en la bibliografía especializada para calcular descomposiciones atómicas. Además, se hará un estudio de los diferentes diccionarios de átomos que se emplean en función de la finalidad de la descomposición atómica a implementar.

- **Desarrollo y metodología de la investigación.**

  Este bloque temático está compuesto de cuatro capítulos, donde se explica el modo en que se ha procedido en la investigación para ir alcanzando los objetivos planteados. Este capítulo constituye el núcleo de la tesis y en él se recogen las principales contribuciones originales.

  En el quinto capítulo de esta tesis doctoral se realiza un estudio detallado del modelado sinusoidal. La principal aportación, entre otras, en este modelado se centra en la extracción tonal guiada perceptualmente con un criterio de parada psicoacústico.

  En el sexto capítulo se estudian las aportaciones realizadas en el modelo de transitorios. Aquí se describen dos modelos de transitorios basados en el algoritmo *matching pursuits*, uno con un diccionario wavelet packets, y otro con un diccionario mixto de wavelet packets y exponenciales complejas.

En el séptimo capítulo, centrado en el modelo de ruido, se explican las herramientas de predicción lineal que tienen en cuenta el comportamiento logarítmico en frecuencia del oído humano, conocidas comúnmente como *warped-lpc*.

Para completar este bloque, en el octavo capítulo se presenta la estructura general del codificador propuesto, detallando la segmentación adaptativa del eje temporal y el proceso de cuantificación de parámetros con principios psicoacústicos. Se detallan los resultados subjetivos, que se han obtenido atendiendo a la recomendación ITU-R BS.1534 (conocida como metodología MUSHRA) para la evaluación subjetiva de medias a grandes degradaciones en los sistemas de audio. Estos resultados permiten comparar la calidad perceptual de las señales decodificadas con las obtenidas usando el estándar MPEG-AAC y el codificador paramétrico estandarizado PPC.

- **Conclusiones y líneas futuras.**

  Este bloque se compone de dos capítulos. En el primero de ellos (capítulo noveno) se presentan las conclusiones obtenidas de la investigación llevada a cabo. Se realiza una revisión de las aportaciones originales introducidas en cada uno de los modelos de señal utilizados en el campo de la codificación paramétrica de audio.

  El siguiente capítulo, que es el décimo y final de la tesis, está dedicado a presentar nuevas líneas de investigación que han surgido durante el desarrollo de la investigación y que suponen el inicio de nuevas vías de investigación en el campo del tratamiento digital de audio, de las cuales pueden derivarse futuras tesis doctorales.

1.4. Principales contribuciones

Finalmente, en este primer capítulo, se presentan las principales contribuciones originales del trabajo de investigación desarrollado:

1. Definición de una nueva medida de importancia perceptual de cada tono en el algoritmo *matching pursuits* con diccionario de exponenciales complejas que permite en cada iteración la extracción de la frecuencia psicoacústicamente más importante (sección 5.2).

2. Definición de un criterio de parada en el algoritmo *matching pursuits* con diccionario de funciones exponenciales complejas que permite detener el algoritmo cuando no quedan en el residuo tonos que estén por encima del umbral de enmascaramiento (sección 5.2).

3. Desarrollo de un algoritmo de codificación de las amplitudes de los tonos que permite enviar un número variable de bits por tono, de forma que cada tono se cuantifica con los bits necesarios para que la cuantificación sea perceptualmente transparente. Este resultado se consigue haciendo que tanto codificador como decodificador calculen de forma sencilla un umbral de enmascaramiento que determine los bits de cada tono (sección 5.3).

4. Inclusión de un proceso rápido de actualización de las correlaciones en el algoritmo *matching pursuits* con un diccionario wavelet packets, basado en las propiedades de las funciones wavelet packets ortogonales. Este algoritmo se emplea en el modelado de transitorios (sección 6.2).

5. Desarrollo de un método de actualización de las correlaciones en el algoritmo *matching pursuits* con un diccionario mixto de funciones wavelet packets y exponenciales complejas, basado en las propiedades de las funciones wavelet packets ortogonales, así como en las propiedades de la transformada discreta de Fourier de las funciones exponenciales complejas. Este algoritmo es idóneo para el modelado de transitorios (sección 6.3).

6. Desarrollo de un modelado de ruido basado en predicción lineal con frecuencia logarítmica (*warped-lpc*) para las frecuencias y en predicción lineal (*tns, time noise shaping*) para el tiempo (sección 7).

7. Desarrollo de un nuevo algoritmo de segmentación flexible del eje temporal (sección 8.2).

Capítulo 2

Introducción a la codificación perceptual de audio

2.1. Necesidad de la codificación de audio

La codificación perceptual de audio digital ha sido a lo largo de los últimos 20 años un campo de aplicación del procesado de señales. Durante este tiempo, se han resuelto algunos de los retos asumidos. Sin embargo, la creciente demanda de aplicaciones digitales en redes telemáticas hace, que aún hoy, la codificación de audio sea un tema de actualidad. El objetivo de este capítulo es describir las técnicas de codificación más utilizadas en el mercado y presentar una revisión del estado del arte en codificación perceptual de audio.

Durante los últimos años, gracias a los esfuerzos de estandarización, ha habido una explosión de aplicaciones, tanto profesionales como de consumo, que han llevado a que el audio digital se haya extendido de forma que se utiliza con asiduidad en la vida cotidiana. Baste, para comprobar este hecho, con enumerar una serie de campos de aplicación:

- Almacenamiento en discos ópticos y dispositivos portátiles.
- Audio asociado para vídeo digital.
- Transmisión de audio mediante redes digitales, por ejemplo internet o redes móviles.
- Radiodifusión digital: DAB (radiodifusión terrestre), WorldSpace (radiodifusión por satélite).

Pese a que el ancho de banda global disponible para la transmisión de señales de audio (y video) aumenta continuamente, así como la capacidad de los dispositivos de almacenamiento, siguen surgiendo campos de aplicación donde los actuales estándares de codificación no ofrecen una solución satisfactoria. En este sentido cabe destacar la necesaria reducción del régimen binario para la transmisión de audio por internet, o telefonía móvil, manteniendo una alta calidad, lo cual ha provocado el desarrollo de la codificación paramétrica de audio. Pero, en un futuro próximo, tal y como adelanta MPEG, van a seguir apareciendo nuevas aplicaciones para el tratamiento digital de audio como, por ejemplo, la búsqueda basada en contenido.

2.2. Requisitos de los sistemas de codificación de audio

A la hora de definir un sistema de codificación de audio, es necesario tener en cuenta los requisitos que se le piden. Dependiendo de la aplicación, algunos de ellos serán más relevantes que otros. Los principales criterios que se tienen en cuenta a la hora de diseñar un esquema de codificación perceptual son los siguientes:

- Eficiencia de compresión. En muchas aplicaciones, obtener la mayor tasa de compresión para la misma calidad de servicio se traduce directamente en ahorro de costes. Por tanto, la calidad de señal para una tasa binaria dada (o la tasa binaria necesaria para conseguir una cierta calidad de señal) es un criterio de diseño importante.

- Calidad de la señal decodificada. En algunas aplicaciones se precisa calidad transparente (no existe diferencia audible entre la señal original y la señal decodificada) o casi transparente. Para asegurar esta calidad el sistema de codificación debe superar pruebas de calidad muy exigentes. En otras aplicaciones, sin embargo, se permite que una persona entrenada distinga la señal original de la decodificada, aunque las distorsiones en la señal decodificada sean tolerables, se habla entonces de audio de alta calidad.

- Complejidad. Para aplicaciones de consumo, la complejidad de la codificación, y en especial de la decodificación, es importante, aunque conforme pasa el tiempo estos aspectos están pasando a un segundo plano. Podemos distinguir distintos tipos de complejidad:

  - Complejidad computacional. Se refiere al número de instrucciones del procesador necesarias para tratar un bloque de muestras. Si el algoritmo de codificación se implementa en una arquitectura de cálculo de propósito general (PC o estación de trabajo), esta es la figura de complejidad más importante.
  - Requisitos de almacenamiento. Supone un factor de coste importante para realizaciones con dispositivos portátiles o bien en arquitecturas de propósito específico (DSP's).
  - Complejidad del codificador frente a la del decodificador. En la mayoría de los algoritmos que se describen en este capítulo, el codificador es más complejo que el decodificador. Esta asimetría es interesante para aplicaciones como la radiodifusión, donde existe una relación de uno a muchos entre el codificador y los decodificadores.

- Retardo. Dependiendo de la aplicación, el retardo puede ser o no un criterio importante. Mientras que es muy importante en aplicaciones donde se dan comunicaciones bidireccionales, no lo es tanto en aplicaciones de almacenamiento. Para radiodifusión, un retardo de 100 ms parece ser tolerable.

- Editabilidad. Desde el punto de vista de codificación el requisito de editabilidad esta relacionado con el de *break-in*, y consiste en la posibilidad de comenzar la decodificación en cualquier punto de la secuencia de bits sin que esto suponga un elevado tiempo de sincronización. Como norma general, un codificador empieza a decodificar antes si no utiliza codificación diferencial entre tramas, puesto que en caso contrario la espera para disponer de todos los valores puede alargarse en el tiempo.

- Resistencia a errores. Dependiendo de la estructura de la secuencia de bits transmitida, los codificadores perceptuales son más o menos sensibles a errores puntuales o de ráfaga producidos en el canal de transmisión. Esta sensibilidad depende del uso que se haga de la codificación diferencial entre diferentes tramas de audio. Evidentemente, la utilización de códigos correctores de errores es una solución, que se consigue a costa de aumentar el régimen binario, la complejidad y el retardo del sistema.

2.3. Codificación perceptual

La función tasa-distorsión determina el régimen binario mínimo que se puede conseguir para una distorsión dada [Berg71]. Normalmente se consiguen muy buenos resultados combinando la eliminación de redundancia (datos que pueden reconstruirse a partir de los presentes), con la eliminación de datos que no son importantes (eliminación de irrelevancia).

La codificación perceptual se centra en la eliminación de aquellos datos que son irrelevantes para el sistema auditivo. La señal se codifica de forma que la distorsión debida a la codificación no sea audible o, por lo menos, en que la distorsión que se produzca sea mínima para el régimen binario objetivo. Para tener éxito en esta tarea, es preciso aplicar el conocimiento disponible sobre el funcionamiento del sentido del oído.

El mínimo régimen binario necesario para codificar una señal de audio sin que se produzcan diferencias perceptuales entre la señal decodificada y la original es la *Entropía Perceptual* (PE) [Johnston88]. La unidad de medida es *bit/muestra*, y se define a partir de la expresión (2.1):

$$PE = \frac{1}{N} \sum_{f=f_l}^{f=f_u} max\Big(0, log_2(\frac{S(f)}{umbral(f)})\Big) \tag{2.1}$$

donde f_l es la frecuencia límite inferior (por ejemplo, $f_l = 0$ Hz), f_u es la frecuencia límite superior (por ejemplo, $f_l = 22{,}050$ kHz), N es el número de componentes frecuenciales entre f_l y f_u, $S(f)$ es la densidad espectral de potencia de la señal y $umbral(f)$ es el umbral de enmascaramiento estimado para dicha señal (el umbral de enmascaramiento se define en la siguiente sección).

Los diferentes codificadores perceptuales han de estimar el umbral de enmascaramiento, lo cual es un paso similar en todos los prototipos. Sin embargo, las diferentes propuestas difieren en cómo obtener los datos de la señal antes de su cuantificación. En cualquier caso, debido a que el umbral de enmascaramiento se define en frecuencia, es necesario realizar una transformación de los datos a este dominio para poder realizar la cuantificación teniendo en cuenta principios psicoacústicos. En función del tipo de transformación que se realice a los datos de la señal de entrada, los codificadores de audio se suelen clasificar en dos grupos principales:

Codificadores por transformada. Se agrupan aquí todos los codificadores que realizan una transformación lineal de la señal de entrada antes de su codificación. Estos codificadores son también conocidos en la bibliografía como codificadores de forma de onda, del inglés *waveform coding*. En general, esta transformación se realiza mediante un banco de filtros o transformada. Se han utilizado un sinfín de transformadas, siendo las más usadas las transformadas de coseno y las transformadas *wavelet-packets*. En los codificadores más avanzados

se adapta el banco de filtros o transformada a las características de la señal de entrada, pudiéndose cambiar incluso en cada trama en que se divide el audio. Las limitaciones de esta forma de codificación se encuentran cuando se quiere reducir demasiado el régimen binario. En este caso, este tipo de codificación no proporciona resultados satisfactorios.

Codificadores paramétricos. Esta forma de codificación surge cuando es necesario reducir mucho el régimen binario. La solución se basa en la modelización de la señal de audio en componentes, los cuales son típicamente: tonos, transitorios y ruido. Un codificador paramétrico extrae parámetros de la señal que modelan estas componentes antes de realizar el proceso de cuantificación. El inconveniente de esta técnica son los errores intrínsecos al modelo, por lo que no es posible conseguir calidad transparente aún cuando se aumente mucho el régimen binario.

2.4. Fundamentos de psicoacústica

La ciencia que estudia las relaciones entre los estímulos acústicos y las sensaciones auditivas se conoce como psicoacústica. En esta sección se introducen los principios en los que se basan los modelos perceptuales que utilizan los modernos codificadores de audio. Estos modelos se aplican para saber cómo cuantificar un determinado valor y que el efecto producido no sea audible en la señal final. Se comenzará haciendo una breve exposición acerca del funcionamiento del sistema auditivo humano. Posteriormente, se analizan las sensaciones auditivas de intensidad sonora, tono y timbre; así como el umbral absoluto de audición, las bandas críticas y las propiedades y tipos de enmascaramiento. Una revisión más detallada de psicoacústica se puede encontrar en [Zwicker90] y en [Moore97].

2.4.1. El sistema auditivo humano

El sistema auditivo humano es la base de la cadena de actuaciones que se realizan en un codificador de audio. Por esta causa, es de vital importancia tener un completo conocimiento de cómo funciona este sistema a la hora de diseñar un codificador de audio. Es esta sección se describe el funcionamiento físico del oído humano, dejando para más adelante las propiedades derivadas que se usan en codificación de audio.

El oído humano (ver figura (2.1)) se puede dividir en tres partes, cada una de las cuales realiza un procesamiento diferente de los sonidos que llegan al sistema:

Oído externo: Es la parte visible del sistema auditivo formado por el pabellón y el canal auditivo. La principal función se limita a la localización de las fuentes del oído en el espacio aunque también realiza otras acciones. Por ejemplo, protege al tímpano del posible daño causado por cuerpos extraños y cambios en la humedad y temperatura. Desde el punto de vista acústico, el canal auditivo (de 2 o 3 cm aprox.) tiene una frecuencia de resonancia cercana a 4 kHz, lo que provoca una ganancia en la señal en este rango de frecuencias, siendo la causa de la alta sensibilidad del oído en esta banda y del mínimo del umbral absoluto de audición [Yost85].

Oído medio: Comienza en el tímpano e incluye toda la cadena de huesos del oído. Básicamente realiza una transmisión del sonido desde el tímpano, a través de los huesos del oído (martillo,

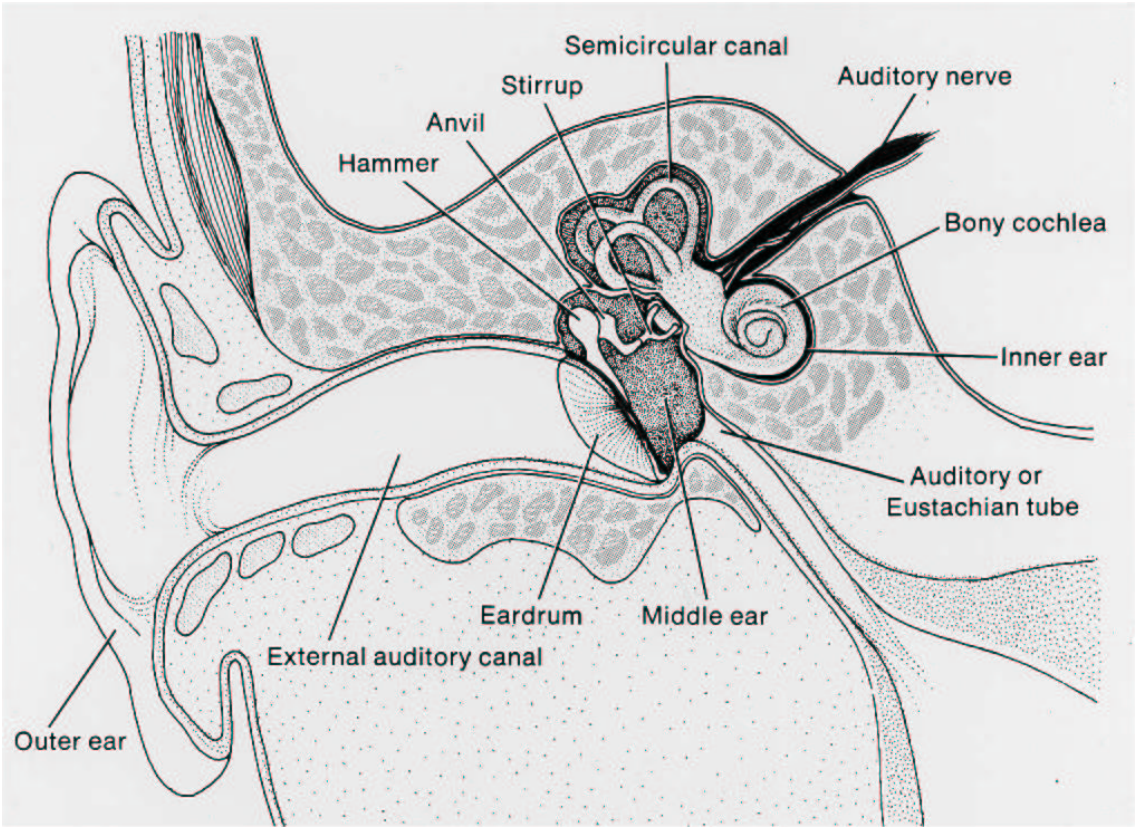

Figura 2.1: *Estructura interna del oído humano. Esta figura se ha obtenido de la dirección de Internet* http://www.owlnet/rice.edu/~psyc351.

yunque y estribo) hasta la entrada del caracol. Esta parte del oído tiene una respuesta adaptada a las frecuencias medias (de 500 a 4000 Hz) porque la adaptación mecánica de las ondas sonoras desde aire (tímpano) a fluido (cóclea) está fisiológicamente diseñada para estas frecuencias.

Oído interno: Es la parte más importante del sistema auditivo desde el punto de vista psicoacústico. Incluye la cóclea o caracol del oído donde se realiza la conversión de señal mecánica a eléctrica. El fluido de la cóclea es excitado por el hueso estribo y estas ondas se propagan hasta donde se encuentran las células sensoriales. Esta propagación tiene la particularidad de que, dependiendo de la frecuencia, el pico de la respuesta de las ondas se sitúa en una parte u otra de la membrana donde están los receptores. Como consecuencia, se excitan diferentes receptores en función de la frecuencia del sonido, de forma que los receptores están sintonizados a la frecuencia de entrada gracias a la conversión frecuencia a lugar que realiza la cóclea. Desde un punto de vista de señal, la cóclea se comporta como un conjunto de filtros paso banda, con anchos de banda no uniformes que crecen con la frecuencia. El concepto de bandas críticas se relaciona con este fenómeno [Zwicker90]. Otro fenómeno que tiene lugar en el oído interno es el enmascaramiento, el cual es producido por la presencia en la misma banda (para los mismos receptores) de una excitación suficiente para bloquear la recepción de una señal más débil. Finalmente, la percepción de un sonido se realiza en el cerebro mediante la composición de las diferentes respuestas eléctricas de las células sensoriales de cada banda enviadas por medio del nervio auditivo.

2.4.2. Umbral absoluto de audición

El umbral absoluto de audición o umbral de silencio indica el nivel de presión sonora (Sound Pressure Level, SPL) en función de la frecuencia en el que un tono puro se empieza a escuchar [Zwicker90]. La figura 2.2 muestra este umbral dependiendo de la frecuencia. Se puede observar cómo el oído es más sensible en el rango de frecuencias de 1 a 5 kHz, principalmente debido a la acción del oído externo. El umbral crece rápidamente tanto en baja como en alta frecuencia. La dependencia de este umbral con la frecuencia fue estudiada por Fletcher [Fletcher40] y aproximada por Terhardt [Terhardt79] mediante la expresión (2.2).

$$T_q(f) = 3,64(\frac{f}{100})^{-0,8} - 6,5e^{-0,6(\frac{f}{1,000}-3,3)^2} + 10^{-3}(\frac{f}{1,000})^4(dB \quad SPL) \tag{2.2}$$

Esta curva es de gran utilizad en codificación de audio porque las componentes frecuenciales bajo este umbral no pueden ser escuchadas y, por lo tanto, no necesitan ser transmitidas. Generalmente, en los codificadores por transformada se eliminan las bandas de señal bajo este umbral. El umbral absoluto de audición se usa en codificación de audio con cautela. En primer lugar, este umbral está asociado a tonos puros, mientras que el ruido de cuantificación en codificadores por transformada no tiene características tonales. En segundo lugar, hay que resaltar que no se tiene conocimiento a priori sobre los niveles reales de reproducción del sonido, aunque como referencia se suele igualar el tono que ocupe todo el rango dinámico del sistema a una intensidad sonora de 96 dB SPL.

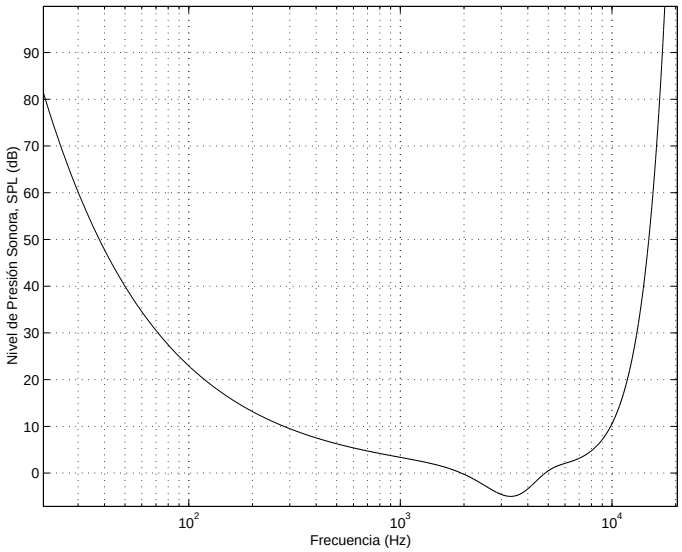

Figura 2.2: *Umbral absoluto de audición*

2.4.3. Intensidad sonora, tono y timbre

La **intensidad sonora** es un atributo de los sonidos en función del cual se pueden ordenar en una escala de más bajo a más alto en intensidad. Además de la potencia de un sonido la intensidad sonora depende también de la duración y la estructura en tiempo y frecuencia del mismo. En el caso de la frecuencia del sonido se definen contornos de la misma intensidad sonora, donde se toma la frecuencia de 1 kHz como referencia. La unidad en la que se mide la intensidad sonora es el fono (*phon*). El umbral de silencio es un ejemplo de contorno de igual intensidad sonora, notar que la intensidad sonora para 1 kHz en el umbral de silencio equivale a 3 fonos. En la figura 2.3 se representan las curvas de igual intensidad sonora partiendo del umbral de silencio.

Otra sensación auditiva es el **tono** que se define como la propiedad que permite ordenar los sonidos en una escala musical. Con el tono se aprecia el patrón de repetición de un sonido, así para el caso de un tono puro se relaciona con su frecuencia, y si se trata de un complejo armónico con la frecuencia fundamental del mismo. En cualquier caso, este atributo es más complejo porque supone que el sonido es armonioso [Moore97]. La asignación de un determinado tono a un sonido significa que se escucha de forma similar a (en la misma escala musical que) la frecuencia de un tono puro.

El **timbre** es otra sensación auditiva, aunque no se puedan ordenar los sonidos en función del timbre en una escala unidimensional. La definición de timbre es negativa, es la propiedad por la cual dos sonidos se distinguen como diferentes aunque tengan la misma intensidad sonora y el mismo tono. En otras palabras, el timbre permite distinguir entre la misma nota tocada, por ejemplo, por un piano y una flauta. Desde un punto de vista psicoacústico, el timbre se detecta en el cerebro al estudiar la composición de señales en diferentes bandas críticas.

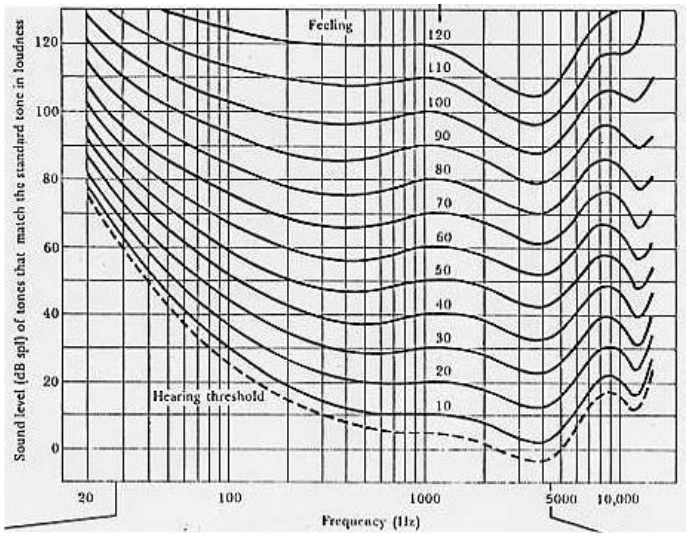

Figura 2.3: *Contornos de igual intensidad sonora para tonos puros. Esta figura se ha obtenido de la dirección de Internet* http://www.owlnet/rice.edu/~psyc351.

2.4.4. Bandas críticas

Como se ha visto, las ondas acústicas que viajan por la cóclea generan picos de respuesta en posiciones específicas de la membrana basilar (donde se encuentran los receptores auditivos) para cada componente frecuencial [Greenwood90]. Como consecuencia de esta transformación, la cóclea se entiende desde el punto de vista del procesado digital de señales como un banco de filtros muy solapados. Las respuestas en amplitud son asimétricas y dependientes del nivel de señal. Además, el ancho de banda, conocido como *ancho de la banda crítica*, no es uniforme y se incrementa con la frecuencia.

La noción de banda crítica se basa en dos hechos experimentales:

1. La intensidad sonora percibida de una fuente de ruido de banda estrecha de nivel constante permanece invariable mientras se incrementa el ancho de banda hasta alcanzar el ancho de la banda crítica, pasado el cual aumenta.

2. El umbral de detección de ruido de banda estrecha que se presenta entre dos tonos enmascaradores permanece constante mientras la diferencia de frecuencia de los tonos se mantiene dentro del ancho de la banda crítica.

El ancho de las bandas críticas permanece aproximadamente constante (unos 100 Hz) hasta los 500 Hz, y se incrementa en aproximadamente un 20 % de la frecuencia central por encima de los 500 Hz. En promedio, el ancho de las bandas críticas puede aproximarse por la expresión (2.3) [Zwicker90], la cual está dibujada en la figura 2.4:

$$BW_c(f) = 25 + 75[1 + 1,4(f/1{,}000)^2]^{0,69}(Hz) \tag{2.3}$$

Resulta usual el tratamiento del oído como un conjunto discreto de bancos de filtros, cuyos anchos de banda se corresponden con los de las bandas críticas. En ese caso, la distancia entre dos bandas críticas adyacentes se conoce normalmente como *un Bark*.

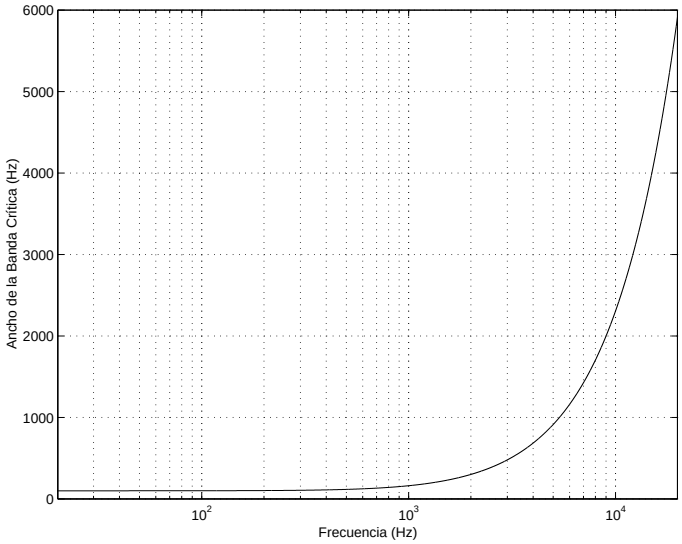

Figura 2.4: *Ancho de las bandas críticas en función de la frecuencia central de la banda.*

2.4.5. Enmascaramiento

En psicoacústica el efecto más importante que debe ser tenido en cuenta en codificación de audio es el enmascaramiento. Se conoce como enmascaramiento el proceso por el cual el umbral de audición de un sonido crece por la presencia de otro sonido. Hay dos tipos diferentes de enmascaramiento: el enmascaramiento simultáneo y el enmascaramiento temporal.

Enmascaramiento simultáneo

Dentro de los estudios sobre psicoacústica, es muy importante el concepto de *enmascaramiento simultáneo*, que describe el efecto mediante el cual una señal débil pero audible (*señal enmascarada o "maskee"*) se hace inaudible cuando otra señal más fuerte (*señal enmascaradora o "masker"*) ocurre de forma simultánea.

La figura 2.5 muestra el umbral de enmascaramiento obtenido a partir del umbral de silencio y del efecto de enmascaramiento producido por dos tonos puros localizados en 1 kHz y 4 kHz. Todas las señales con un nivel de presión sonora por debajo del umbral resultante y que sean simultáneas a estos dos tonos no serán audibles.

En el cálculo del umbral de enmascaramiento debe contemplarse la dispersión del efecto enmascarador hacia las bandas próximas a la de la señal enmascarante. Esta dispersión viene caracterizada por la *Función de Dispersión*, cuya pendiente es más abrupta hacia las bajas que hacia las altas frecuencias. Esta función realiza el efecto de filtrado paso banda que ocurre en la cóclea. Una buena estimación de esta pendiente hacia las bajas frecuencias es de 31 dB/Bark. Por su parte, la pendiente de la función de dispersión hacia las altas frecuencias depende, además, del nivel de presión sonora del elemento enmascarador. Así, elementos enmascaradores de mayor intensidad producen un mayor enmascaramiento hacia las altas frecuencias (una pendiente más suave de la función de dispersión). Valores de -6 dB/Bark para señales de alta intensidad y de -10 dB/Bark para señales de menor intensidad se citan en [Zwicker90]. Mientras que en [Terhardt79] se aproxima por la expresión (2.4):

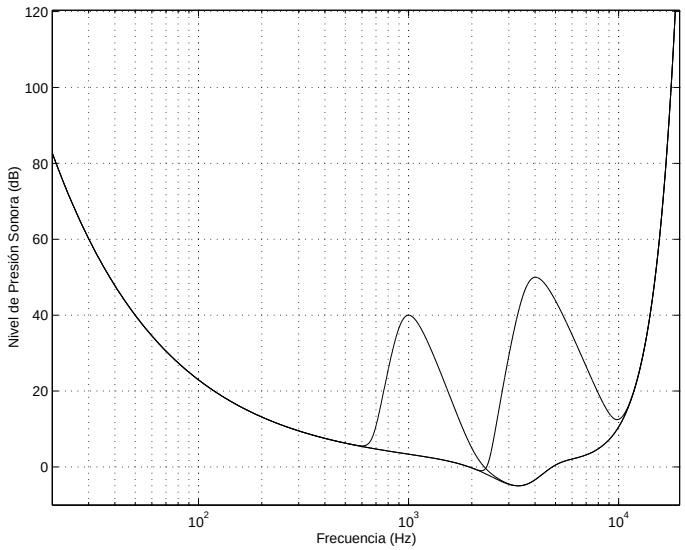

Figura 2.5: *Efecto de enmascaramiento de dos tonos en 1kHz y 4kHz*

$$22 + min(\frac{230}{f}, 10) - 0,2L \quad (dB/Bark) \tag{2.4}$$

donde f es la frecuencia del tono enmascarador y L su intensidad en dB SPL.

Finalmente, hay que destacar que la capacidad de enmascaramiento depende de la tonalidad del elemento enmascarador. Un ruido de banda estrecha presenta una mayor capacidad de enmascaramiento sobre un tono que al contrario [Hell72]. Esta propiedad se conoce como asimetría en el enmascaramiento. La forma de tratar con esta característica del enmascaramiento en la bibliografía [Zwicker90] es tener dos tipos de señales y dos efectos, resultando cuatro escenarios de enmascaramiento: ruido que enmascara a tonos (noise-masking-tone, NMT), tonos que enmascaran a tonos (tone-masking-tone, TMT), tonos que enmascaran a ruido (tone-masking-noise, TMN), y ruido que enmascara a tonos (noise-masking-noise, NMN).

- NMT: El nivel de enmascaramiento en la misma banda crítica depende del nivel de ruido que enmascara, así en [Hall98] se dice que el umbral se sitúa a 4 dB con 80 dB SPL de ruido enmascarador y a 3 dB para 60 dB SPL.

- TMT: Cuando tanto señal enmascaradora como enmascarada son tonos se dan [Hall98] 19 dB para un tono enmascarador de 400 Hz de 80 dB SPL, 15 dB para 60 dB SPL, y 14 dB para 40 dB SPL. Otros valores se pueden encontrar en [Zwicker90] en función de la frecuencia, pero son pocos los estudios realizados porque este tipo de enmascaramiento no se utiliza en codificadores por transformada. El efecto enmascarador de un tono es más fuerte si la duración del tono es mayor hasta un máximo de 300 ms [Par02], lo que puede ser tenido en cuenta por codificadores paramétricos. Además, en este caso se han observado situaciones especiales, porque cuando ambos tonos están muy próximos en frecuencia tienden a interferirse y provocar fluctuaciones en la intensidad sonora [Lee03].

- TMN: La máscara generada en este caso depende tanto del nivel de presión sonora del tono como de su frecuencia. Pero, comparada con el caso del ruido enmascarador, un tono

tiene menos capacidad de enmascaramiento. En [Hall98] se presentan los valores para una frecuencia de 1 kHz, así el umbral está a 21 dB para un tono de 60 dB SPL, a 24 dB para 80 dB SPL y a 28 dB para 90 dB SPL. Hay una gran conjunto de expresiones similares [Zwicker90] [Moore97] [MPEG92] para este escenario que aprovechan los diferentes modelos perceptuales que emplean los codificadores de audio.

- NMN: Es un valor difícil de medir en la práctica porque no se puede distinguir entre tipos de ruido. Los valores que aparecen en la bibliografía son muy diversos, así en [Hall98] aparece un valor genérico de 26 dB, mientras que en el modelo de enmascaramiento 2 de MPEG [MPEG92] se utiliza un valor de $5,5$ dB.

El valor de la máscara final se ha de obtener reconociendo cuantas señales enmascaradoras ruidosas y tonales hay en la señal. Esto se consigue de forma general calculando la tonalidad de la señal en cada banda crítica. A partir de este valor se divide la señal de entrada en parte tonal y parte ruidosa dentro de la banda crítica para obtener el umbral de enmascaramiento [MPEG92].

Un aspecto clave aún no resuelto completamente en psicoacústica es la *aditividad del enmascaramiento*. Si existen varios elementos enmascaradores y los efectos de enmascaramiento particulares de cada uno de ellos se solapan, el enmascaramiento combinado es normalmente mayor que el esperado a partir de los cálculos realizados con las energías de las señales [Beer92].

En la mayoría de los casos, los modelos psicoacústicos que utilizan los codificadores de audio se limitan a calcular, a partir de la señal de entrada, el umbral de enmascaramiento simultáneo en frecuencia. Este umbral de enmascaramiento se refiere, en los codificadores por transformada, al ruido de cuantificación que se puede inyectar en una frecuencia dada. Sin embargo, para el caso de los codificadores paramétricos, es recomendable calcular el umbral de enmascaramiento tanto para tonos como para ruido.

Enmascaramiento temporal

El efecto de enmascaramiento de una señal se extiende en el tiempo tanto a instantes previos a la propia generación del elemento enmascarador (*'pre-masking' o enmascaramiento hacia atrás*) como a instantes posteriores a su extinción (*'post-masking' o enmascaramiento hacia delante*) como se puede observar en la figura 2.6. Este efecto hace posible que se puedan usar sistemas de análisis/síntesis con una resolución temporal limitada (por ejemplo, bancos de filtros con gran resolución en frecuencia) para codificar audio digital de alta calidad. Los datos experimentales sugieren que el enmascaramiento hacia atrás presenta una gran variación entre sujetos, así como también entre diferentes señales usadas como elementos enmascaradores y enmascarados.

Las señales indeseadas (*artefactos*) generadas por el codificador que se extienden en el tiempo de forma que preceden a una transición de la señal en el dominio temporal (por ejemplo, un ataque brusco de percusión) pueden dar lugar a distorsiones audibles conocidas como *pre-ecos*. Dado que los codificadores basados en bancos de filtros siempre originan una dispersión temporal del error de cuantificación (en la mayoría de los casos superior a 4 ms), el pre-eco es un problema bastante común en los sistemas de codificación de audio. En la figura 2.7 podemos ver un claro ejemplo de distorsión de pre-eco.

La forma típica de minimizar el efecto de pre-eco es el uso de una segmentación adaptativa del eje temporal cuando la señal cambia su contenido, es decir, que el tamaño de trama de audio

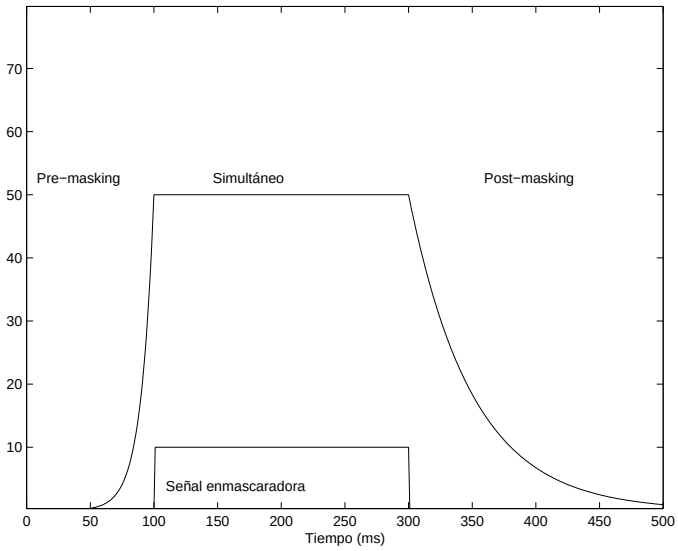

Figura 2.6: *Ejemplo de pre-masking y post-masking*

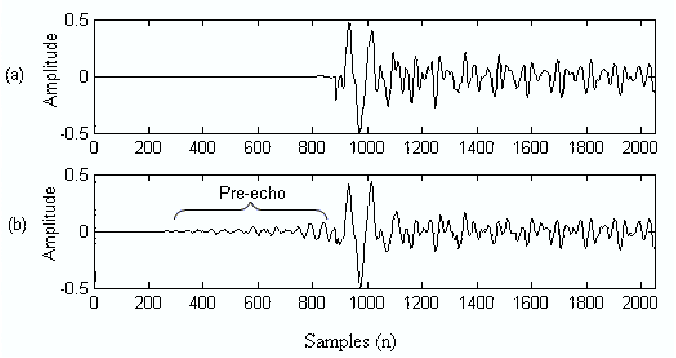

Figura 2.7: *Ejemplo de pre-eco*

sea variable. Teniendo en cuenta los valores de enmascaramiento temporal se pueden establecer las fronteras entre tramas de forma que el efecto de pre-eco sea inaudible.

2.4.6. *Just Noticeable Difference*

Los codificadores paramétricos de audio deben de cuantificar, no las muestras de salida de un banco de filtros, sino todo un conjunto de parámetros dependientes de la señal. Como ejemplo, para el caso de la componente tonal de la señal de audio, los parámetros extraídos son la amplitud, frecuencia y fase de cada tono y, en algunos casos, su duración. Las componentes ruidosa y transitoria tienen sus propios parámetros. Un modelo de enmascaramiento clásico sólo proporciona información de cómo cuantificar las amplitudes de tonos y ruido no teniendo una herramienta válida para el resto de parámetros. Ante este problema se han realizado estudios de la capacidad de discriminación o resolución del oído ante una serie de parámetros tonales, ruidosos o transitorios. Esta resolución se conoce por el nombre de *Just Noticeable Difference* (JND) en la bibliografía.

Por ejemplo, para el caso de la frecuencia de un tono el valor de JND en frecuencia depende de la duración del mismo: $0,2$ Bark para 10 ms y $0,01$ Bark para 500 ms [Zwicker90]. La resolución del oído respecto a otros parámetros se puede encontrar en la bibliografía relacionada con psicoacústica [Zwicker90] [Moore97].

2.5. Elementos básicos de un codificador perceptual de audio

2.5.1. Introducción

El objetivo básico en codificación perceptual de audio digital de alta calidad consiste en ocultar la distorsión producida por la codificación por debajo de la capacidad de enmascaramiento y resolución propias del oído humano. Como la señal de audio es una señal no estacionaria, la primera aproximación consiste en analizar la señal en diferentes segmentos temporales donde las características de la señal sean casi estacionarias. Entonces se estima el umbral de enmascaramiento simultáneo en el dominio de la frecuencia, ocultando el efecto de la cuantificación bajo este umbral. Sin embargo, este enfoque es diferente en un codificador paramétrico que descompone la señal en tonos, transitorios y ruido.

En general, la codificación perceptual de audio se plantea como un análisis tiempo/frecuencia, habiendo dos enfoques principales: 1) el uso de un banco de filtros o transformada en codificación por de forma de onda, y 2) el empleo de un modelo de la señal extrayendo los parámetros de este modelo en codificación paramétrica de audio. Esto conduce a una estructura básica de los codificadores perceptuales que es común a prácticamente todos los sistemas actuales.

La figura 2.8 muestra el diagrama de bloques básico de un sistema de codificación perceptual de audio, cuyos elementos constitutivos son:

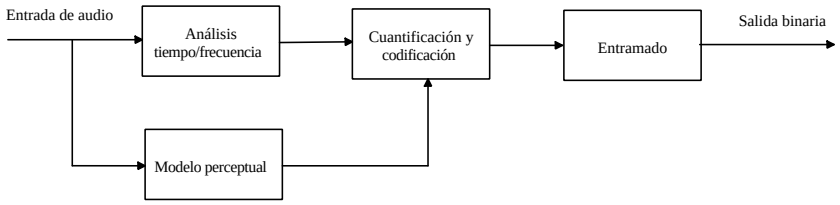

Figura 2.8: *Diagrama de bloques de un sistema de codificación perceptual*

- Análisis tiempo/frecuencia.
- Modelo perceptual.
- Cuantificación y codificación.
- Entramado.

2.5.2. Análisis tiempo/frecuencia

Todos los codificadores de audio utilizan alguna técnica de análisis tiempo-frecuencia para extraer una serie de coeficientes o parámetros a partir de la señal de audio que pueden ser cuantificados y codificados, atendiendo a alguna medida de distorsión perceptual. Como se ha visto

anteriormente, en función del tipo de análisis tiempo/frecuencia, se clasifican los codificadores de audio en dos categorías radicalmente diferentes.

Codificadores por transformada

La herramienta más usada para realizar el análisis tiempo/frecuencia hasta hace pocos años era un banco de filtros, el cual descompone la señal en sub-bandas de frecuencia. Este banco de filtros juega un papel importante en la determinación de irrelevancias cuando se usa conjuntamente con un modelo perceptual. La figura 2.9 muestra el diagrama de bloques básico de un banco de filtros de análisis/síntesis de n canales con un factor de diezmado de k.

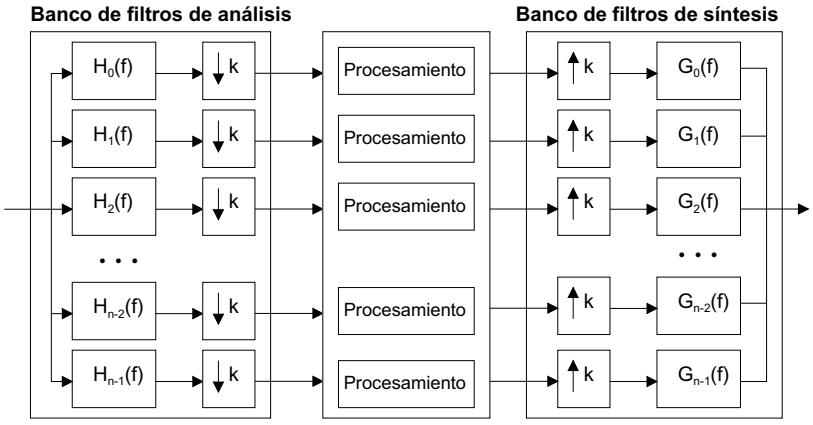

Figura 2.9: *Diagrama de bloques de un banco de filtros de análisis/síntesis*

El diseño del banco de filtros debe perseguir el objetivo general de representar la señal de entrada con el menor número de bits posible. Deben tenerse en cuenta varios aspectos de diseño:

1. La descomposición debe ser invertible, es decir, el banco de filtros debe ser de reconstrucción perfecta o casi perfecta. Esta propiedad es muy importante para asegurar que la distorsión en la señal reconstruida es debida al proceso de cuantificación.

2. Tanto los filtros de análisis como los de síntesis deben ser muy selectivos en frecuencia, con objeto de que la aplicación de la información psico-acústica sea lo más simple posible.

3. El número de componentes espectrales por unidad de tiempo debe ser lo más bajo posible. Para ello se suelen usar sistemas con muestreo crítico, donde el número de componentes espectrales es igual al número de muestras de la señal en el dominio temporal.

4. Se suele decir que el ancho de banda de los filtros del banco debe ser menor o igual que el ancho de la banda crítica más estrecha, porque así se facilita el control de la percepción del ruido de cuantificación. Esta aseveración no es rigurosamente cierta, porque si la descomposición subbandas se adapta a la descomposición en bandas críticas, sólo es preciso asegurar que el ancho de cada subbanda sea menor o igual que el ancho de la banda crítica más próxima.

5. Además, el banco de filtros no debe dispersar el ruido de cuantificación más allá de una ventana temporal lo suficientemente amplia como para asegurar que el umbral de enmascaramiento permanece invariable y de esta forma evitar los problemas de pre-eco.

6. El coste computacional es otro factor importante. Los filtros IIR se implementan con un bajo coste computacional y, además, proporcionan alta selectividad, pero desafortunadamente es difícil implementar bancos de filtros de reconstrucción perfecta usando filtros IIR.

7. Bancos de filtros estáticos o dinámicos. Los errores de cuantificación de las componentes espectrales pueden manifestarse en la señal de salida, extendiéndose en el tiempo sobre la longitud de la ventana de síntesis, dando lugar a distorsiones audibles (pre-ecos). Este efecto indeseable puede reducirse si el banco de filtros no es estático, sino que conmuta entre distintas resoluciones tiempo/frecuencia para los diferentes segmentos de audio.

Entre los tipos de bancos de filtros que se han venido utilizando en los sistemas de codificación perceptual de audio, podemos citar los siguientes:

1. Bancos de filtros QMF.

2. Bancos de filtros que implementan descomposiciones wavelet.

3. Bancos de filtros polifásicos. Se trata de bancos de filtros con ancho de banda uniforme que combinan la flexibilidad de diseño de los bancos QMF con una baja complejidad computacional. La mayoría de los diseños actuales se basan en [Rothweiler83]. Su principal inconveniente es que no permiten obtener descomposiciones no uniformes en frecuencia (todas las subbandas tienen la misma anchura).

4. Bancos de filtros basados en cancelación del solapamiento temporal (TDAC). Dentro de ellos, destaca la Transformada Discreta del Coseno Modificada (MDCT) [Princen87], que se puede interpretar como el enfoque dual de los bancos QMF con cancelación del solapamiento frecuencial. Combina muestreo crítico, buena resolución en frecuencia y alta eficiencia computacional. Normalmente, se emplean realizaciones que van desde 128 a 2.048 bandas igualmente espaciadas. La transformada MDCT también es conocida como transformada solapada modulada (MLT) [Malvar90].

5. Bancos de filtros híbridos. Son aquellos que constan de una sucesión de diferentes tipos de bancos. Se propusieron inicialmente [Brandenburg90] para conseguir un sistema de análisis/síntesis que combinara la posibilidad de obtener diferentes resoluciones en frecuencia a distintas frecuencias con estructuras QMF en árbol y la eficiencia computacional de los algoritmos del tipo FFT.

Sin embargo, en paralelo con el perfeccionamiento y estandarización de la codificación que emplea bancos de filtros para analizar la señal, han surgido otros codificadores optimizados para trabajar a bajo régimen binario y que utilizan otras herramientas para el análisis tiempo/frecuencia.

Codificadores paramétricos

En la compresión de diferentes fuentes de información se buscan sistemas que incluyan un modelo de generación de la fuente, con objeto de reducir la cantidad de datos necesaria para enviar la señal de forma fidedigna. Siguiendo este principio, en el caso de la señal de voz, existen codificadores que alcanzan una extraordinaria tasa de compresión. Es importante reseñar que en este tipo de codificadores de voz no se representa la forma de onda, sino la forma de producción de la señal. Estos valores de tasa de compresión no son alcanzables cuando se trabaja con codificadores por transformada para señales de audio.

La existencia de este modelo de producción de la señal de voz ha permitido el desarrollo de numerosas aplicaciones, no sólo de codificación, sino de reconocimiento de voz y de locutores, basándose casi siempre en la medida de las diferencias entre los parámetros del modelo de producción o en parámetros alternativos obtenidos a partir de ellos.

Desgraciadamente, en el caso de la señal de audio, este modelo de producción no es adecuado, muy al contrario, no es posible establecer un modelo de generación basado en principios físicos, como en el caso de la señal de voz, debido a la diferente naturaleza de las señales que forman el audio en general. Como consecuencia, las técnicas de codificación estandarizadas, por muy sofisticadas que sean, se engloban dentro de los codificadores por de forma de onda.

Sin embargo, es posible extraer parámetros de la señal de audio dividiendo la señal en sus componentes: tonos, transitorios y ruido. Utilizando herramientas que obtengan parámetros de estas componentes es posible la obtención de un modelo de señal adaptativo para el audio. El desarrollo de este modelo para las señales de audio musicales permite no sólo conseguir altas tasas de compresión en aplicaciones de codificación, sino que es el punto de partida en aplicaciones de clasificación de señales, descripción de la información multimedia, e indexado, que serán ampliamente demandadas en un futuro inmediato dentro del sector multimedia.

Así pues, en un codificador paramétrico el análisis tiempo/frecuencia se realiza extrayendo de la señal los parámetros correspondientes a cada una de las componentes de señal:

Tonos La extracción tonal se conoce con el nombre de modelo sinusoidal y la forma más sencilla de implementarla es la detección de picos espectrales en el dominio de Fourier. Esta parte de la señal representa las características de la señal de audio que cambian lentamente con el tiempo.

Transitorios Se modela en esta componente los breves incrementos de energía que se producen en las señales de audio, como golpes de castañuelas o batería. Los algoritmos utilizados en la bibliografía son muy diversos para el tratamiento de esta componente.

Ruido En los sistemas actuales, el residuo de señal que no se puede modelar como tonos o transitorios se trata como ruido. En cualquier caso, esta componente representa la parte estocástica de la señal de audio. Es reseñable que algunos instrumentos, como la flauta, generan un flujo de señal ruidosa de forma adicional a la señal tonal.

La figura 2.10 muestra la descomposición de un golpe de batería en sus componentes: tonos, transitorios y ruido. En este ejemplo la componente transitoria es importante, algo que ocurre con poca frecuencia en la señal de audio. Así, durante la mayor parte de la duración de la señal basta con descomponerla en tonos y ruido. Debido a que la hipótesis de la investigación desarrollada

se centra en el empleo de codificación paramétrica de audio, en el siguiente capítulo se realizará un estudio profundo del estado del arte en esta materia.

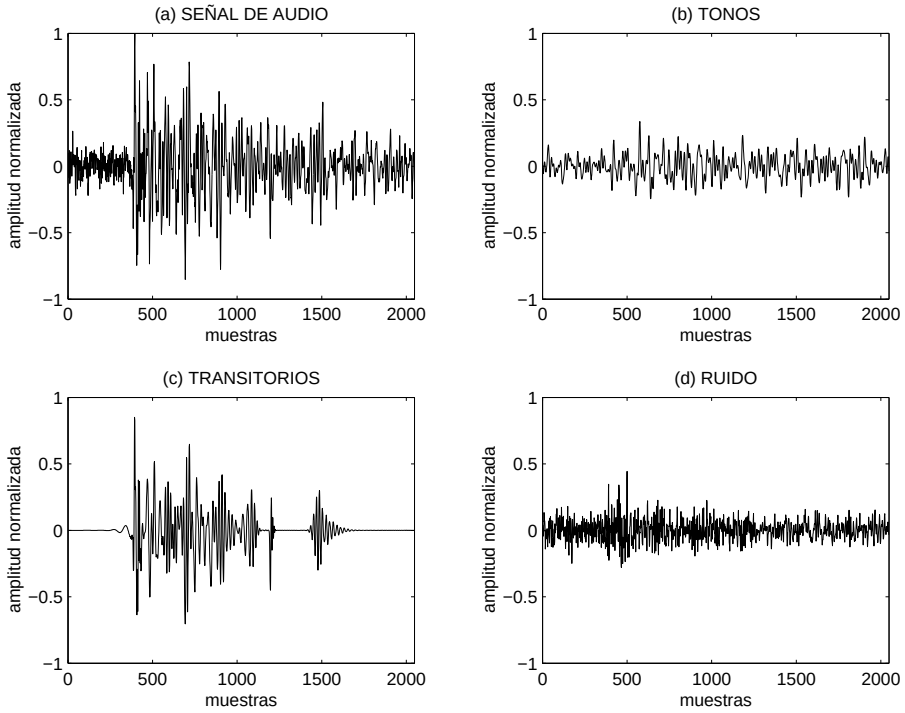

Figura 2.10: *Descomposición de un golpe de batería en sus componentes.*

2.5.3. Modelos perceptuales

La función principal del modelo psico-acústico en un sistema de codificación perceptual de audio es proporcionar estimaciones precisas del máximo ruido permitido (umbrales de enmascaramiento), de acuerdo con la resolución tiempo/frecuencia del sistema de codificación. En muchas ocasiones, como por ejemplo en el estándar MPEG para audio, simplemente se define el formato de transmisión, permitiendo cambios y mejoras en el modelo perceptual.

Debido a las diferentes capacidades de enmascaramiento de las señales tonales y del ruido [Scharf70, Hell72], un cálculo importante dentro de todo modelo psico-acústico o perceptual es el del *índice de tonalidad*. La forma en que se determine dicho índice incidirá de forma fundamental en la eficacia del modelo perceptual.

Entre las diversas formas de cálculo del índice de tonalidad asociado a las distintas componentes espectrales de una señal, merece la pena citar el algoritmo propuesto por Brandenburg [Brandenburg90], por constituir la base de uno de los métodos propuestos en el estándar MPEG para audio. Brandenburg propone el uso de un simple predictor polinómico para calcular el índice de tonalidad. El cálculo se basa en la utilización de dos segmentos anteriores, localizados en $t-1$ y $t-2$, para predecir el módulo $r(t,\omega)$ y fase $\Phi(t,\omega)$ de cada línea frecuencial del segmento localizado en t.

Los valores de predicción $\hat{r}$ y $\hat{\Phi}$ de r y Φ se calculan de la siguiente forma:

$$\hat{r}(t,\omega) = r(t-1,\omega) + (r(t-1,\omega) - r(t-2,\omega)) \tag{2.5}$$

$$\hat{\Phi}(t,\omega) = \Phi(t-1,\omega) + (\Phi(t-1,\omega) - \Phi(t-2,\omega)) \tag{2.6}$$

La distancia euclídea ponderada entre los valores reales y los predecidos define la *impredicibilidad* $c(t,\omega)$, algunas veces denominada *medida de caos*:

$$c(t,\omega) = \frac{dist\{[\hat{r}(t,\omega), \hat{\Phi}(t,\omega)], [r(t,\omega), \Phi(t,\omega)]\}}{r(t,\omega) + |\hat{r}(t,\omega)|} \tag{2.7}$$

Si la componente de pulsación ω de la señal es muy tonal, la predicción será acertada y $c(t,\omega)$ tomará un valor muy pequeño. Por el contrario, si la señal es de tipo ruidoso, $c(t,\omega)$ tomará valores hasta 1 con una media de 0,5. Por tanto, la medida de caos puede ser limitada al rango 0,05 y 0,5, donde para 0,05 la señal es considerada completamente tonal y para 0,5 completamente ruidosa:

$$c_l(t,\omega) = max\{0,05;\ min[0,5, c(t,\omega)]\} \tag{2.8}$$

La medida de caos $c(t,\omega)$ se convierte en índice de tonalidad $v(t,\omega)$ mediante una transformación no lineal:

$$v(t,\omega) = -0,43\ log_{10}(c_l(t,\omega)) - 0,299 \tag{2.9}$$

El índice de tonalidad $v(t,\omega)$ nos da el resultado final de la estimación de tonalidad, que puede ser aplicada a un modelo perceptual, como por ejemplo el modelo perceptual 2 del estándar MPEG-1, el cual es uno de los modelos más ampliamente utilizados en los codificadores perceptuales actuales. Se puede encontrar una breve descripción del mismo en [Kahrs98], tal y como se extrae del anexo informativo del estándar MPEG-1 audio [MPEG92].

Sin embargo, en los últimos años se ha desarrollado algún modelo de enmascaramiento que no utiliza un índice de tonalidad propiamente dicho, aprovechando los últimos avances en psicoacústica. Así, en [Par02] se utiliza la propiedad del sistema auditivo humano de integrar la distorsión presente en un conjunto de filtros auditivos (o bandas críticas). En este sentido, el índice de tonalidad deja de tener sentido, pues una señal ruidosa genera más distorsión en el oído porque abarca mayor número de filtros auditivos que una señal tonal que sólo afectará a alguno de ellos. Básicamente, el esquema del modelo de enmascaramiento presentado en [Par02] se dibuja en la figura 2.11, donde se incluye el ruido interno del oído que es el que limita la sensibilidad de las señales más débiles o umbral de silencio.

2.5.4. Cuantificación y codificación

La etapa de cuantificación y de codificación juega un papel muy importante en el sistema de codificación perceptual. Se pueden considerar un gran número de opciones de diseño, tanto para la cuantificación como para la posterior codificación de las muestras o parámetros cuantificados:

- Alternativas de cuantificación:

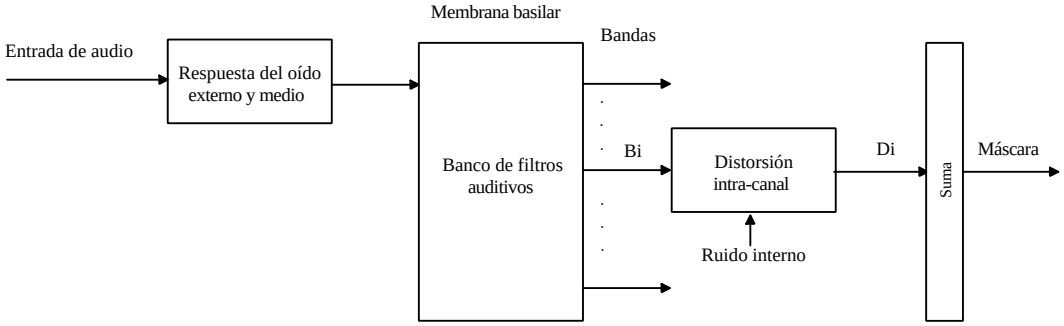

Figura 2.11: *Esquema de un modelo de enmascaramiento sin índice de tonalidad.*

1. *Cuantificación escalar uniforme.*
2. *Cuantificación escalar no uniforme.* Se aplica normalmente para reducir la potencia de ruido de cuantificación y eliminar la correlación entre valores cuantificados cuando el número de escalones de cuantificación es reducido. El cuantificador óptimo en este sentido es el Max-Lloyd.
3. *Cuantificación vectorial.* En este caso no se cuantifican valores individuales, sino agrupaciones de éstos. Se usa en la mayoría de los esquemas actuales de codificación de voz e imagen, pero ha sido poco utilizada en codificación de audio. Un ejemplo de aplicación con éxito de cuantificación vectorial es el sistema denominado TWIN-VQ [Iwakami95], propuesto como parte del estándar MPEG-4 audio [MPEG97b].

- Alternativas de codificación:

  Los valores cuantificados se almacenan y/o transmiten, bien directamente mediante una estrategia de asignación de bits (incluyendo *bit packing*), o bien como palabras código resultantes de una etapa de codificación entrópica.

- Estructuras de control para cuantificación y codificación:

  1. *Asignación de bits (estructura directa).* En este caso, un algoritmo de reparto decide cuántos bits se asignan a cada muestra o parámetro, atendiendo bien a parámetros estadísticos de los datos o bien a un modelo perceptual. Este proceso se realiza antes de que se efectúe la cuantificación.
  2. *Asignación de ruido (estructura indirecta).* Esta es una estrategia sólo aplicable a codificadores por transformada. Las muestras se cuantifican mediante modificaciones del tamaño del escalón de cuantificación atendiendo a un modelo perceptual. El número de bits asignados a cada valor no se conoce hasta que el proceso de asignación de ruido se ha completado.

2.6. Estándares en codificación de audio

En esta sección se presenta una revisión de algunos esquemas de codificación de audio que se han propuesto a lo largo de los años, centrándose en los estándares propuestos por MPEG

(Moving Pictures Experts Group) desde su creación en 1988 hasta la fecha. Existen otros sistemas comerciales de codificación de audio, que han tenido cierta importancia, pero no se han incluido en esta revisión, como por ejemplo AC-2 y AC-3 [Todd94], competidores directos, respectivamente, de las capas 2 y 3 del estándar MPEG-1 audio.

Se puede encontrar una revisión bastante completa de los sucesivos estándares MPEG audio en [Brandenburg97, Painter00]. Hasta la fecha han sido cuatro los estándares internacionales de codificación de audio desarrollados por el grupo MPEG: MPEG-1 audio, MPEG-2 audio, MPEG-2 AAC y MPEG-4 audio.

2.6.1. MPEG-1 Audio - capas 1 y 2

El estándar MPEG-1 audio fue propuesto en 1992 [MPEG92]. Se diseñó como respuesta a la necesidad de múltiples aplicaciones: almacenamiento de audio digital en cintas magnéticas, radio digital, transmisión de audio mediante RDSI, etc. Se ideó un sistema de codificación estructurado en tres modos de funcionamiento, crecientes en complejidad, a los que se llamó capas. La capa 1 fue inicialmente optimizada para un régimen binario de 192 kbits/s por canal (se empleó en el *Digital Compact Cassette, DCC*), la capa 2 para un régimen binario de 128 kbits/s por canal y la capa 3 para 64 kbits/s por canal. Se permiten tres frecuencias de muestreo de 32 kHz, 44,1 kHz y 48 kHz.

Las características fundamentales de las capas 1 y 2 son las siguientes:

1. Para agrupar los datos de entrada en bloques se usa un algoritmo de segmentacion fija.

2. Se usa un banco de filtros polifásico que convierte la entrada de audio digital en 32 subbandas. Utiliza un filtro prototipo de 511 coeficientes. Para cada subbanda de salida es preciso transmitir la siguiente información:

   - Asignación de bits. Determina el número de bits empleados para codificar las muestras de cada subbanda. En la capa 1 se usan 4 bits, mientras que en la capa 2 existen diferentes patrones para enviar dicha información, dependiendo del régimen binario deseado y de la frecuencia de muestreo.
   - Factores de escala. El cálculo de los factores de escala se realiza cada 12 muestras de subbanda. Sólo se transmiten los factores de escala correspondientes a bloques de muestras con asignación binaria distinta de cero.
   - Muestras en la subbanda. Las muestras de cada subbanda se transmiten usando la longitud de palabra definida por el algoritmo de asignación de bits. Se emplea cuantificación uniforme y cuantificadores en huella.

3. La asignación de bits se realiza a partir de los resultados proporcionados por un modelo psicoacústico. El cuantificador utilizado está basado en compansión de bloques y se añade, además, un codificador de trama.

A diferencia de la capa 1, la capa 2 del estándar MPEG-1 usa una longitud de trama de 36 muestras. Mientras la información de asignación de bits es válida para toda la trama, los factores de escala se actualizan cada 12 muestras, como ocurre en la capa 1. En la capa 2 se utilizan 2

bits por subbanda y trama para determinar si se transmite uno, dos o tres factores de escala por trama.

Mientras que en la capa 1 las posibles asignaciones de bits son 0 y de 2 a 15 bits, en la capa 2 se permiten de forma adicional cuantificadores de 3, 5, 7 y 9 niveles de cuantificación, lo que conlleva una considerable disminución del régimen binario.

2.6.2. MPEG-1 Audio - capa 3

La capa 3 combina algunas de las características de la capa 2 con una mayor eficiencia en la codificación, que se consigue gracias a una mayor resolución en frecuencia y a la utilización de codificación Huffman estática, tal y como se propone en el sistema ASPEC [Brandenburg91]. La figura 2.12 muestra el diagrama de bloques correspondiente a la capa 3 del sistema MPEG-1 audio.

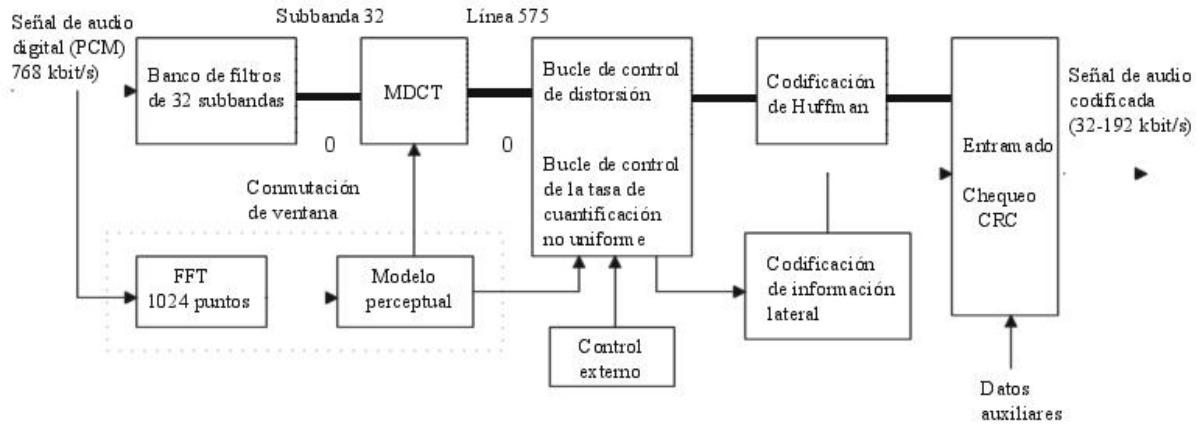

Figura 2.12: *Diagrama de bloques del esquema de codificación MPEG-1 audio capa 3*

Entre las novedades incorporadas en la capa 3 destacan:

1. Banco de filtros híbrido conmutado, que admite tres posibles variantes correspondientes a resoluciones en frecuencia de 576, 216 y 192 líneas.
2. Cuantificación no uniforme.
3. Control del ruido de cuantificación mediante análisis por síntesis.
4. Codificación Huffman de los valores cuantificados.

2.6.3. MPEG-2 Audio

El estándar MPEG-2 audio (ISO/IEC 13818-3) presenta dos grandes avances en relación al estándar MPEG-1 audio (ISO/IEC 11172-3):

Codificación multicanal compatible hacia atrás

El estándar internacional MPEG-2 audio contiene la definición de un sistema de codificación multicanal compatible hacia atrás, denominado *MPEG-2 BC*. Los canales L y R del estándar

MPEG-1 son sustituidos por las señales L_C y R_C, definidas por las expresiones (2.10) y (2.11), y codificadas mediante un sistema MPEG-1 audio.

$$L_C = \frac{1}{1 + 1/\sqrt{2} + a} \left[L + C/\sqrt{2} + a\ L_S \right] \tag{2.10}$$

$$R_C = \frac{1}{1 + 1/\sqrt{2} + a} \left[R + C/\sqrt{2} + a\ R_S \right] \tag{2.11}$$

El coeficiente a puede tomar alguno de los siguientes valores: $\frac{1}{\sqrt{2}}, 1/2, \frac{1}{2\sqrt{2}}, 0$. En consecuencia, un decodificador MPEG-1 audio puede reproducir con buena calidad una versión *downmix* partiendo de una señal multicanal.

En MPEG-2 BC el formato de trama básico es idéntico al formato de MPEG-1. Los canales adicionales (C, L_S y R_S, para el caso de 5 canales) se transmiten en el campo de datos auxiliares de la estructura de trama MPEG-1. En la figura 2.13 puede apreciarse cómo se realiza la transmisión de la información multicanal MPEG-2 dentro de la estructura de trama MPEG-1.

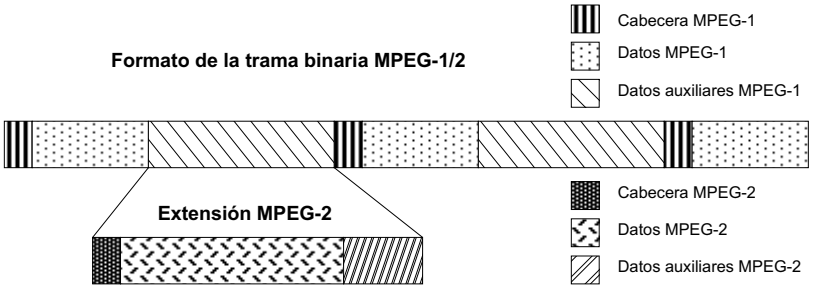

Figura 2.13: *Estructura de trama MPEG-1 para la transmisión de información multicanal MPEG-2*

Otra novedad del estándar MPEG-2 audio es la adición de nuevos modos de funcionamiento que emplean frecuencias de muestreo más bajas (por debajo de 32 kHz). Estos modos son útiles para las siguientes aplicaciones: transmisión de voz de banda ancha y audio de calidad media a regímenes binarios comprendidos entre 16 y 64 kbits/s por canal, transmisión de voz en aplicaciones de comentarista, sistemas de audio por Internet, o cualquier otra aplicación donde la cantidad bits a repartir sea un recurso muy escaso.

MPEG-2 Advanced Audio Coder (MPEG-2 AAC)

Está recogido en la norma ISO/IEC 13818-7. Se trata de un nuevo estándar de codificación de audio que no es compatible hacia atrás, adecuado para configuraciones flexibles de canal y que introduce servicios estéreo y multicanal.

Se organiza en un conjunto de herramientas de codificación que se pueden seleccionar dependiendo de la CPU disponible, de los recursos del canal y de la calidad deseada, de entre tres perfiles de complejidad. Cada uno de ellos recomienda una combinación de herramientas. Las herramientas principales son:

1. Banco de filtros basado en la transformada discreta de coseno modificada (MDCT). La resolución temporal depende de la señal a analizar, oscilando entre 2.048 muestras para

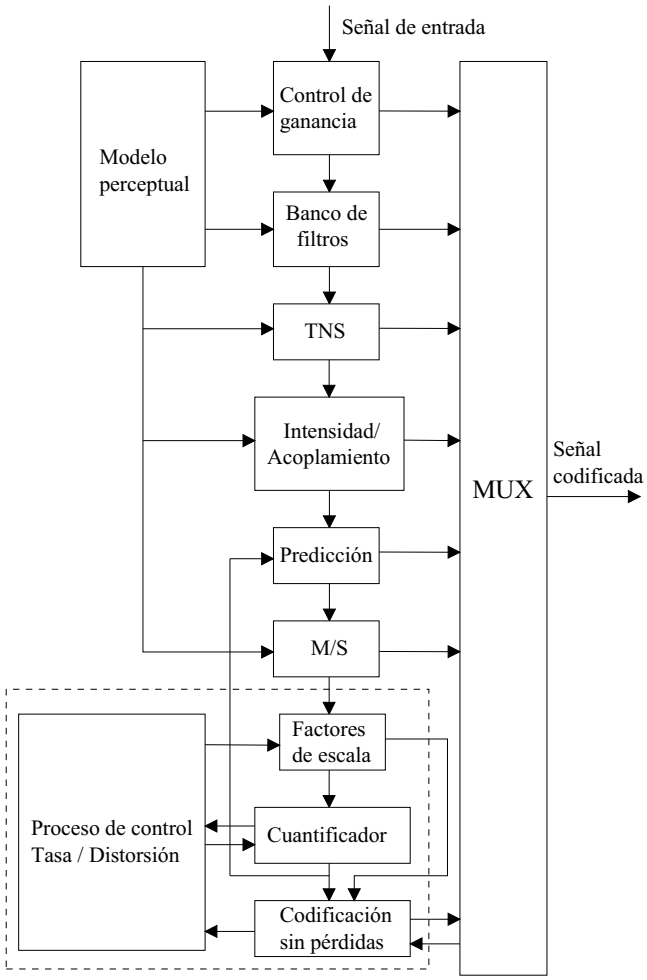

Figura 2.14: *Diagrama de bloques del estándar de codificación MPEG-2 AAC*

señales estacionarias y 256 muestras durante los transitorios. Se puede elegir entre dos formas de ventana alternativas: una ventana sinusoidal cuando sea más conveniente la selectividad que la atenuación en la banda eliminada (señales con estructura armónica densa, es decir, con armónicos próximos en frecuencia) y una ventana KBD cuando se requiera una alta atenuación en la banda eliminada.

2. Conformación temporal del ruido (Time Noise Shaping, TNS). Controla la forma temporal del ruido de cuantificación dentro de cada ventana de la transformada para minimizar la distorsión de pre-eco.

3. Predicción, para disminuir la redundancia en señales estacionarias.

4. Factores de escala. El espectro se divide en varios grupos de coeficientes espectrales (subbandas) que comparten un parámetro común denominado factor de escala. Un factor de escala representa un valor de ganancia que se usa para modificar la amplitud de todos los coeficientes espectrales contenidos en la subbanda correspondiente. Este proceso conlleva una conformación del ruido de cuantificación, de acuerdo con los umbrales de enmascaramiento

estimados por el modelo perceptual.

5. Cuantificación. Se usa cuantificación no uniforme (como en la capa 3 del estándar MPEG-1 audio).

6. Codificación sin pérdidas. Se aplica codificación Huffman estática para el espectro cuantificado, los factores de escala diferenciales y la información direccional. Se emplean un total de 12 tablas Huffman estáticas para codificar agrupaciones de dos o cuatro valores espectrales.

7. Se emplea un modelo psicoacústico similar al modelo 2 del estándar ISO/IEC 11172-3.

2.6.4. MPEG-4 Audio

El estándar MPEG-4, cuya designación formal es ISO/IEC 14496 [MPEG98], no es sólo un estándar de codificación de audio sino más bien un conjunto de herramientas, basadas en descripciones estructuradas, que cubren un amplio conjunto de aplicaciones en audio, voz, audio sintético, texto-a-voz, etc. El estándar incluye, además, una serie de herramientas asociadas como son: escalabilidad , procesamiento de efectos especiales, manipulaciones en los sonidos, y composiciones 3-D, entre otras. Como consecuencia, MPEG-4 proporciona los elementos tecnológicos que hacen posible la integración de los paradigmas de producción, distribución y acceso al contenido en los campos de televisión digital, aplicaciones gráficas interactivas y multimedia interactivo, además de satisfacer las necesidades de autores, proveedores de servicios de red y usuarios finales.

Como todos los estándares MPEG anteriores, MPEG-4 se subdivide en diferentes partes siendo una de ellas la codificación de audio. En esta parte, MPEG-4 incluye una gran variedad de aplicaciones que van desde la codificación inteligible hasta la codificación de alta calidad multicanal, o desde los sonidos naturales a los sintéticos [Koenen99]. Así pues, MPEG-4 audio estandariza la codificación de sonidos naturales a regímenes binarios que van desde 2 hasta 64 kbits/s, incluso cuando se permite codificación con régimen variable, es posible trabajar con tasas binarias inferiores a 2 kbits/s. A continuación, se describen brevemente las herramientas y perfiles que proporciona el estándar MPEG-4.

Voz Para la codificación de voz, MPEG-4 proporciona dos codificadores HVXC (Harmonic Vector eXcitation Coding) y CELP (Code Excited Linear Predictive) que funcionan a distinto rango de régimen binario y calidad:

- La codificación HVXC para un rango recomendado de 1,2 a 4 kbits/s por canal con una frecuencia de muestreo de 8 kHz. Se trata de un codificador paramétrico que descompone la señal en tonos con relación armónica, componentes tonales individuales y ruido; y que, debido a esta organización, permite un cambio de pitch y/o de velocidad directo en el decodificador. Notar que, para conseguir bajar por debajo de 2 kbits/s es necesario funcionar en régimen variable.
- La codificación CELP para un rango recomendado de 4 a 24 kbits/s por canal. Admite dos frecuencias de muestreo de 8 y 16 kHz para voz de banda estrecha y de banda ancha, respectivamente.

Voz sintética MPEG-4 incorpora un interfaz de texto-a-voz TTS (Text-To-Speech) que soporta regímenes binarios en el rango de 200 bits/s hasta 1,2 kbits/s, permitiendo como entrada texto o texto con parámetros prosódicos (valor del pitch, duración del fonema, etc.) para generar voz sintética de calidad inteligible. El algoritmo para obtener la voz sintética no está especificado en el estándar, sólo se define el interfaz de entrada. Además se incluyen las siguientes funcionalidades [Koenen02]:

- Síntesis de audio usando los parámetros prosódicos de la voz original.
- Funcionalidades como pausa, espera, salto hacia delante o hacia atrás.
- Soporte para lenguajes internacionales y dialectos.
- Símbolos internacionales para fonemas.
- Especificación de la edad, género y velocidad de habla del hablante.
- Especificación de parámetros asociados a animación facial FAP (Facial Animation Parameter).

Audio sintético En relación a la generación de sonidos sintéticos, MPEG-4 audio define decodificadores para sintetizar audio basados en varias clases de entradas estructuradas. Estos decodificadores hacen uso de un lenguaje de síntesis especial llamado SAOL (Structured Audio Orchestra Language) para generar música sintética. MPEG-4 no estandariza un único método de síntesis musical, sino que describe varios métodos de síntesis. Cualquier método actual o futuro de síntesis puede ser descrito en SAOL.

Audio MPEG-4 utiliza AAC para la codificación de audio de banda ancha. Sin embargo, se han definido nuevas funcionalidades con el objetivo de conseguir una señal codificada de alta calidad a muy bajo régimen binario, por debajo de los 64 kbits/s por canal del diseño original de AAC. El resultado de la inclusión de estas nuevas herramientas permite una nueva propiedad en la codificación de audio, la escalabilidad, es decir, la capacidad de producir representaciones de la señal codificada con un régimen binario escalable. Es importante reseñar, que esto significa que es posible decodificar subconjuntos de los datos codificados, o eliminar partes del mismo, durante la transmisión sin necesidad de volver a codificar la señal. Así, dependiendo de las características particulares de la conexión, un terminal puede recibir sólo subconjuntos de los datos de entrada y ser capaz de decodificar una señal de buena calidad. Las nuevas técnicas incluidas en MPEG-4 que permiten escalabilidad son:

- Long Term Predictor (LTP): Esta herramienta se ha incluido para evitar la alta complejidad computacional del predictor definido originalmente en MPEG-2 AAC, aunque consigue una ganancia de codificación similar. Como en un codificador de voz, el predictor LTP se implementa en el dominio del tiempo antes del banco de filtros basados en la transformada de coseno MDCT. La señal resultado del predictor se resta a la original en el dominio de la frecuencia permitiendo de esta forma la aplicación de esta herramienta sólo en las bandas seleccionadas.
- Perceptual Noise Substitution (PNS): Esta herramienta realiza la sustitución de algunas bandas de frecuencia por ruido aleatorio. La técnica se basa en la propiedad

del oído por la que la percepción subjetiva de una señal de naturaleza ruidosa depende de la envolvente de energía temporal y espectral, no de la verdadera forma de onda, lo que permite una reducción significativa del régimen binario en las bandas en las que se aplique. El resultado es la aplicación de un modelo paramétrico de ruido [Herre98], que en lugar de cuantificar las todas muestras de la banda en cuestión, sólo procesa aquellos coeficientes que definen la envolvente espectral (y temporal). El funcionamiento resumido es el siguiente: si en una banda de frecuencias se detecta una señal ruidosa, se estiman los coeficientes para modelar la envolvente, típicamente un filtro todo polos, y se envían al decodificador los coeficientes y un indicador para informar que se va a realizar la sustitución. En el decodificador se genera ruido aleatorio que se filtra mediante los coeficientes recibidos para obtener la envolvente deseada.

- Twin VQ: Durante el proceso de estandarización se recibieron dos codificadores de audio que superaban al resto de propuestas, ambos basados en la transformada de coseno MDCT. Uno de ellos era AAC, que tenía mejores resultados subjetivos a alto régimen binario, mientras que el otro codificador era el conocido como Twin VQ que obtenía mejores resultados a bajo régimen binario (por debajo de 16 kbits/s). Sin embargo, al utilizar la misma transformada, la ventaja de Twin VQ radica en el empleo de un esquema de cuantificación y codificación alternativo, descrito en [Iwakami95], y basado en algoritmos derivados de la cuantificación vectorial. Finalmente, se ha incluido en el estándar esta técnica como una herramienta a utilizar cuando el régimen binario requerido esté por debajo de 16 kbits/s.

Como resultado de la inclusión en el estándar de esta serie de objetos, se puede representar la señal de voz o audio de forma eficiente eligiendo adecuadamente las herramientas a utilizar en función de la aplicación, como se destaca en la figura 2.15. Adicionalmente, para permitir esta utilización eficiente, el estándar define subconjuntos de herramientas que se agrupan para utilizarlas en aplicaciones concretas. Estas agrupaciones se conocen como *profiles* y se utilizan para delimitar el conjunto de herramientas a integrar en un decodificador concreto. En la versión 1 del estándar se definen las siguientes:

- *Speech*: se incluyen el codificador paramétrico de voz HVXC, el codificador de voz CELP y el interfaz de voz sintética (de texto a voz) TTS.
- *Scalable*: se aplica para codificación escalable de voz y música en redes de telecomunicación, e incluye el profile anterior, más todas las herramientas de codificación de audio natural.
- *Synthesis*: es un interfaz para generar sonidos y voz sintéticos a muy bajo régimen binario, e incluye todas las herramientas de voz y audio sintético definidas en el estándar.
- *Main*: contienen todas las herramientas definidas en el estándar.

Para finalizar, MPEG-4, en su versión 1, ofrece una serie de funcionalidades en el decodificador entre las que se pueden destacar las siguientes:

- Escalabilidad de régimen binario, de ancho de banda y de complejidad en el codificador y decodificador.

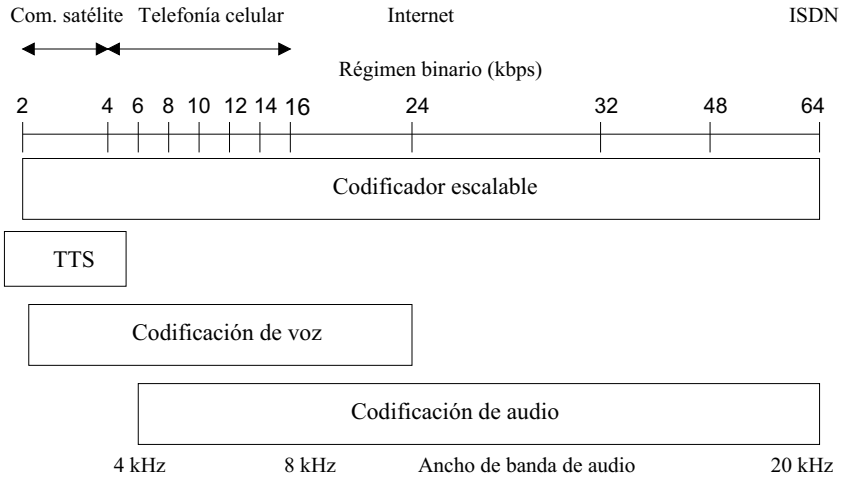

Figura 2.15: *Aplicaciones del estándar MPEG-4 audio*

- Efectos de audio: mezcla, reverberación, etc.
- Cambio de velocidad. Permite cambiar la escala temporal sin alterar el *pitch* durante el proceso de decodificación.
- Cambio de *pitch*. Permite cambiar el *pitch* sin alterar la escala temporal durante la codificación o decodificación. Se aplica sólo a métodos de codificación estructurados y paramétricos.

En 1999 el grupo de estandarización MPEG aprobó la versión 2 de MPEG-4, concebida como una extensión y, por tanto, compatible hacia atrás con la versión 1. En esta extensión se incluyen nuevas herramientas, no para reemplazar a las anteriores, sino para añadir nuevas funcionalidades [Purnhagen99b] como:

- Nuevos métodos de protección contra errores para canales de alta probabilidad de error. Por ejemplo, reordenación de las palabras código de Huffman para AAC.
- Codificación de audio de bajo retardo, pensada para comunicaciones bidireccionales en tiempo real.
- Escalabilidad granular del régimen binario.
- Codificación paramétrica de audio, lo que permite modificar la escala temporal y el *pitch* durante la decodificación sin la necesidad de una unidad de procesamiento de efectos. Este codificador paramétrico se conoce como HILN (Harmonic and Individual Lines plus Noise) [Purnhagen00] y divide la señal de audio en tres partes como se observa en la figura 2.16. Dos partes se extraen de la componente tonal del audio, una para los tonos con relación armónica y otra para los que no poseen esta propiedad, mientras que la parte restante modela el comportamiento ruidoso de la señal de audio. Por tanto, no hay un tratamiento adecuado para los transitorios en HILN, por lo que no se puede hablar de un codificador de alta calidad.

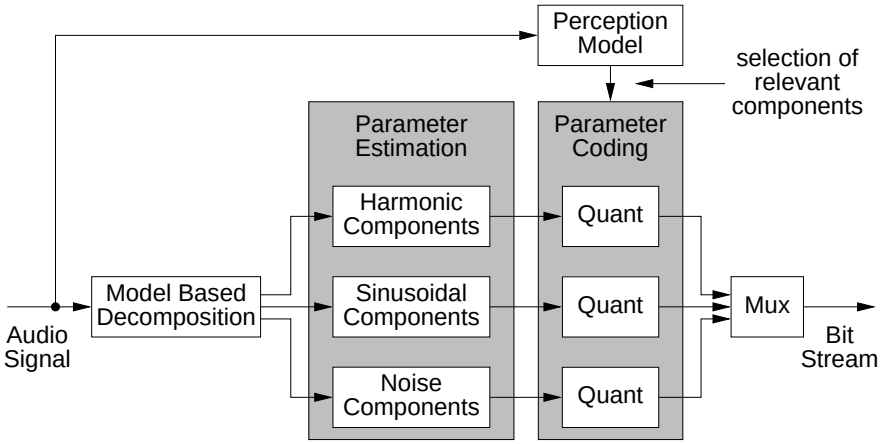

Figura 2.16: *Diagrama de bloques del codificador paramétrico HILN [Purnhagen00].*

- Parametrización de las propiedades acústicas de una escena MPEG-4, lo que permite la presentación de sonido 3D mejorado, modelado acústico del recinto, etc.

En esta versión 2 de MPEG-4 se añaden además una serie de *profiles* a la lista ya definida en la versión 1. Los nuevos subconjuntos de herramientas son:

- *High quality*: para codificación de señales con alta calidad; incluye el codificador de voz CELP y para audio AAC de baja complejidad con LTP.
- *Low delay*: para codificación de bajo retardo; incluye todas las herramientas disponibles en el estándar con estas características, como son los codificadores de voz HVXC y CELP, y el codificador de bajo retardo de AAC de la versión 2 de MPEG-4.
- *Natural audio*: contiene todas las herramientas disponibles para codificación de audio natural sin incluir las de audio sintético.
- *Mobile audio internetworking*: pensado para codificación de audio a bajo régimen binario; incluye AAC con herramientas asociadas como, por ejemplo, bajo retardo o TwinVQ.

En 2001 el grupo MPEG hizo una llamada a propuestas [MPEG01] con el objeto de encontrar nuevos desarrollos tecnológicos que permitieran mejoras en el estándar. El resultado ha sido la adopción en MPEG-4 de nuevas herramientas que extienden el estándar, con lo que surge MPEG-4 versión 3, o MPEG-4 extensión 1 como también se conoce a este conjunto de nuevas funcionalidades. Las herramientas incluidas más destacadas son:

- Extensión del ancho de banda. La replicación de bandas espectrales (*Spectral Band Replication*, SBR) fue propuesta por la empresa *Coding Technologies* y adoptada por el estándar MPEG-4 [Dietz03]. Esta técnica asume que las bandas espectrales de alta frecuencia perdidas en una señal de audio por efecto del filtrado se pueden recuperar a partir de la señal paso bajo y una pequeña cantidad adicional de información de control. Una descripción más detallada de la tecnología SBR se puede encontrar en [Ziegler02] o [Dietz02].

- Codificación paramétrica de audio de alta calidad. La extensión al codificador paramétrico HILN de la versión anterior se centra en la codificación de audio paramétrica de alta calidad. El objetivo inicial de la llamada a propuestas era mejorar la calidad del AAC a 24 kbits/s para todas las señales de prueba. El codificador paramétrico adoptado [MPEG03], propuesto por Philips [Schuijers03] [Kerkhof02] y conocido como PPC (Philips Parametric Coder) está optimizado para la codificación a 24 kbits/s en señales estéreo, permitiendo en el codificador el cambio de pitch y de velocidad de reproducción de forma directa. En este codificador, a diferencia del codificador HILN de la anterior versión de MPEG-4, se integra un modelado específico de los transitorios, algo indispensable para obtener alta calidad para todo el conjunto de señales de prueba. Como se observa en la figura 2.17 la extensión a estéreo del codificador en mono también se hace de forma paramétrica y añade un régimen binario de 0,5 a 7 kbits/s sobre el codificador en mono.

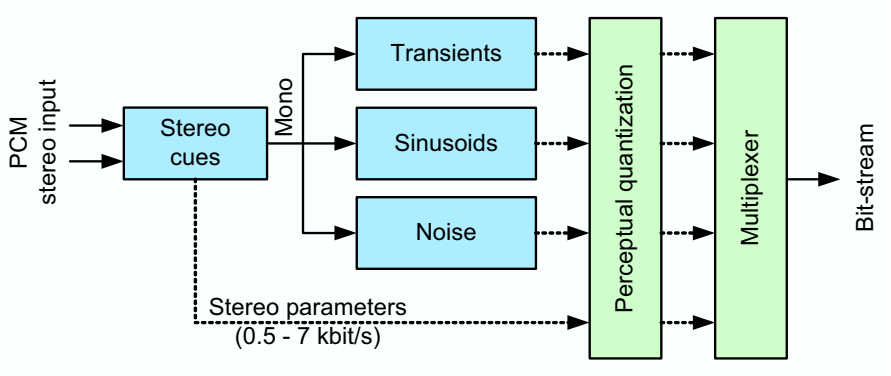

Figura 2.17: *Diagrama de bloques del codificador paramétrico PPC [Schuijers03].*

- Codificación de audio sin pérdidas. Esta nueva herramienta, a diferencia de las dos anteriores centradas en la reducción del régimen binario, se orienta hacia las aplicaciones de alta calidad. Este codificador sin pérdidas de MPEG-4 (conocido como MPEG-4 ALS, Audio Lossless Coding) tiene las siguientes características:

  1. Recuperación sin pérdidas de señales de audio PCM con frecuencias de muestreo comprendidas desde 44,1 kHz hasta 192 kHz, y un número de bits por muestra de 16, 20 y 24 bits.
  2. Mejora en eficiencia de compresión con respecto a cualquier algoritmo conocido hasta el momento.
  3. Proporciona facilidades de edición, manipulación y acceso aleatorio al audio comprimido.

  El esquema de codificación para predicción, que es el corazón del codificador sin pérdidas, está basado en el propuesto por la empresa *RealNetworks* en [Quackenbush03]. Sin embargo, el codificador ALS, con el predictor anterior incluido, fue finalmente estandarizado en 2004 a partir de una propuesta de la Universidad Técnica de Berlín [Liebchen04].

2.7. Calidad perceptual

Un codificador de audio obtiene una buena calidad perceptual cuando la señal codificada es prácticamente indistinguible al escucharla con respecto a la señal original. Esta situación ideal se puede obtener fácilmente si el diseño del codificador atiende a principios perceptuales y, a su vez, se dispone del régimen binario suficiente para poder codificar la señal con los bits que necesita. Sin embargo, esta situación ideal no se puede alcanzar en ciertas aplicaciones donde el ancho de banda disponible es reducido. En estas situaciones es útil poder obtener una medida de la calidad perceptual para saber qué codificador de audio se adapta mejor al requisito de ancho de banda.

La medida de la calidad perceptual se puede realizar mediante dos enfoques radicalmente diferentes. Por un lado, se puede realizar una evaluación objetiva de la calidad. Sin embargo, las medidas objetivas fáciles de implementar, como por ejemplo la relación señal a ruido, no ofrecen una idea de la calidad real que obtiene un codificador basado en principios perceptuales. Es verdad que existen enfoques, como por ejemplo [ITU-R01b], para la implementación de medidas objetivas, pero la implementación de estas recomendaciones es bastante complicada. Además, las medidas objetivas son difícilmente aplicables cuando las degradaciones presentes en la señal codificada son medias o grandes. Por otro lado, la evaluación subjetiva, basada en la opinión de personas que escuchan la señal codificada, puede llegar a ser bastante dispar en función del grupo de oyentes seleccionados, pero tiene la ventaja de poder realizarse sin más medios que el mismo grupo de oyentes.

Para realizar la medida subjetiva de la calidad es bastante crítico elegir el método a utilizar en función de la calidad perceptual del codificador. Así, para codificadores que introducen leves degradaciones en la señal la escala MOS es la más utilizada en la bibliografía. Se engloban aquí la mayoría de los codificadores por transformada a medio-alto régimen binario de funcionamiento. Sin embargo, cuando el codificador introduce mayores degradaciones en la señal de audio codificada se suele utilizar el método MUSHRA para medir la calidad perceptual. Esto sucede en la mayoría de los codificadores paramétricos y en los codificadores por transformada a bajo régimen binario. Para ambos métodos existen recomendaciones del ITU-R que explican de forma detallada cómo realizar las pruebas subjetivas con el fin de obtener unos resultados comparables entre diferentes pruebas de audición.

2.7.1. La escala MOS

La calidad subjetiva de la mayoría de los codificadores perceptuales por transformada se mide en la bibliografía a partir de la escala MOS (*Mean Opinion Score*). Esta escala se obtiene como resultado de pruebas de audición basadas en la recomendación ITU-R BS.1116-1 [ITU-R97]. A continuación, se realiza un breve resumen de los aspectos más relevantes de esta recomendación para la evaluación subjetiva de pequeñas degradaciones en los sistemas de audio. En principio, se expondrán los aspectos que son de aplicación a sistemas monofónicos, dado que el sistema de codificación de audio propuesto en esta tesis está diseñado únicamente para señales de un solo canal.

Cuadro 2.1: *Escala de degradación de 5 notas.*

DEGRADACION	NOTA
Imperceptible	5.0
Perceptible, pero no molesta	4.0
Ligeramente molesta	3.0
Molesta	2.0
Muy molesta	1.0

Selección de los oyentes

Es importante que los datos de las pruebas de escucha para evaluar pequeñas degradaciones en los sistemas de audio procedan exclusivamente de participantes con experiencia en detectar dichas pequeñas degradaciones. Cuanto mayor sea la calidad alcanzada en los sistemas que deben someterse a prueba, más importante será contar con oyentes expertos. Para la selección de los participantes, pueden seguirse dos procedimientos:

- *Selección previa de los participantes.* Este procedimiento incluye métodos tales como pruebas audiométricas, selección de participantes basándose en su experiencia y desempeño en pruebas anteriores.
- *Selección posterior de los participantes.* Los métodos de selección posterior puede ser de dos tipos: uno se basa en las incoherencias respecto al resultado medio y el otro en la capacidad del participante para realizar identificaciones correctas.

En relación al número de oyentes, éste puede estimarse calculando la varianza y determinando la resolución necesaria del experimento. Cuando las condiciones de una prueba de escucha están determinadas por los aspectos técnicos y de comportamiento de los participantes, la experiencia ha demostrado que a menudo bastan los datos procedentes de 20 de ellos para extraer las conclusiones adecuadas de la prueba.

Método de prueba

Para las evaluaciones subjetivas de sistemas que producen pequeñas degradaciones, es necesario seleccionar un método adecuado. Un método especialmente sensible, estable y que permite detectar con exactitud pequeñas degradaciones es el de *triple estímulo doblemente ciego con referencia oculta*. Por consiguiente, es el que se utiliza para este tipo de prueba.

En la forma más adecuada y sensible de este método, sólo actúa un participante cada vez y lo hace seleccionando a discreción uno de entre tres estímulos (A, B, y C). La referencia conocida siempre es el estímulo A. La referencia oculta y el objeto son B y C, asignados de manera aleatoria, dependiendo del experimento. Se solicita al participante que evalúe las degradaciones en B comparadas con las de A, y las de C comparadas también con las de A, de acuerdo con la escala continua de degradación de cinco notas. Uno de los estímulos, B o C, debe ser indistinguible del estímulo A, el otro puede presentar degradaciones. Toda diferencia percibida entre la referencia y los otros estímulos debe interpretarse como una degradación.

La escala de apreciaciones puede considerarse continua con referencias obtenidas de la escala de degradación de cinco notas del ITU-R [1] indicada en la tabla 2.1. Se recomienda utilizar la escala con una resolución de un número decimal. El método de prueba consta de dos partes o fases:

Fase de adiestramiento o familiarización Antes de realizar la apreciación formal, se debe permitir a los participantes familiarizarse con los dispositivos, el entorno de prueba y con el proceso de prueba y las escalas de apreciación. Los participantes deben familiarizarse también con las señales de prueba. Si se lleva a cabo de manera correcta el proceso de familiarización, se puede transformar a algunos participantes con habilidad acústica inicialmente baja en expertos a efectos de la prueba. Al finalizar dicho proceso, los participantes deben haber adquirido un conocimiento preciso de la escala a emplear en la fase de apreciación formal.

Fase de apreciación Como la memoria auditiva a medio y largo plazo no es fiable, el procedimiento de prueba debe basarse exclusivamente en la memoria corto plazo. Para ello, lo más adecuado es utilizar un método de conmutación casi instantánea, que exige una alineación en el tiempo entre los estímulos de aproximadamente 40 ms. Los participantes deben poder actuar de forma individual. Sólo de esta manera tendrán completa libertad para conmutar entre los estímulos. Esta libertad es esencial para realizar comparaciones detalladas entre los estímulos de cada experimento. Es preferible que los participantes puedan conmutar entre los estímulos sin ayuda visual, de forma que, si lo desean, puedan mantener los ojos cerrados para concentrarse mejor. Una sesión de apreciación no debe durar más de 20 o 30 minutos, si bien el carácter de auto-control del ritmo de los experimentos aquí señalado dará lugar a variaciones en la duración de la prueba según los participantes. La experiencia sugiere que no deben programarse más de 10 o 15 experimentos por sesión para lograr la duración de sesión deseada. La fatiga de los participantes puede convertirse en un factor perjudicial que reste validez a sus juicios. Para evitar esta circunstancia, entre sesiones sucesivas de cada participante deben preverse períodos de descanso de duración al menos igual a la de una sesión.

Atributos

En la recomendación UIT-R BS.1116-1b se indican los atributos específicos de las evaluaciones monofónicas, estereofónicas y multicanal. Es preciso evaluar en los tres casos el atributo *calidad de audio básica*. Este atributo sencillo y general se utiliza para juzgar una o todas las diferencias detectadas entre la referencia y el objeto. Este atributo es el evaluado en sistemas de codificación monofónicos. Para los sistemas estereofónicos y multicanal, la recomendación UIT-R BS.1116-1 define otros atributos adicionales, como la *calidad de la imagen estereofónica* (sistemas estéreo) y la *calidad de la imagen frontal* y la *impresión de la calidad panorámica* (sistemas multicanal).

Material de programa

Sólo se utiliza material crítico para poner de relieve las diferencias entre los sistemas sometidos a prueba. No hay un material de programa "adecuado" de forma universal que pueda utilizarse

Cuadro 2.2: *Señales del cd EBU-SQAM.*

Nombre	Señal
es01	Suzanne Vega
es02	German male speech
es03	English female speech
si01	Harpsichord
si02	Castanets
si03	Pitch pipe
sm01	Bagpipes
sm02	Glockenspiel
sm03	Plucked strings
sc01	Trumpet solo
sc02	Orchestra piece
sc03	Contemporary pop

para evaluar todos los sistemas bajo todas las condiciones. En consecuencia, debe encontrarse para cada sistema probado en cada experimento, el material de programa crítico apropiado. Sin embargo, son de uso común en la actualidad ciertas bases de datos propuestas por varios organismos. Es especialmente usada la propuesta por la EBU (*European Broadcasting Union*) para el aseguramiento de la calidad en sistemas de audio [Waters98], recopiladas en el cd EBU-SQAM. Las señales propuestas en esta base de datos para medir la calidad de codificadores de audio son 12 señales y se subdividen en 4 grupos cada uno de 3 señales:

Señales vocales, ya sean habladas o cantadas.

Señales de un sólo instrumento interpretando notas aisladas (*single tone*).

Señales de un sólo instrumento interpretando una melodía (*melodious phrase*).

Señales más complejas. Estas tres señales son una trompeta tocando una melodía, un pieza de orquesta y una señal de pop actual.

Las 12 señales se detallan en la tabla 2.2.

Análisis estadístico

El objeto fundamental del análisis estadístico de los resultados de prueba es identificar con exactitud la calidad de funcionamiento media de cada uno de los sistemas sometidos a prueba y la fiabilidad de cualquier diferencia entre los valores obtenidos. Este último aspecto obliga a efectuar una estimación de la variabilidad o varianza de los resultados. Se recomienda, en condiciones normales, usar del modelo de análisis de varianza (ANOVA). Para realizar un estudio estadístico en detalle habría que considerar otros métodos de análisis, como por ejemplo los no paramétricos.

Presentación de resultados de los análisis estadísticos

La presentación debe realizarse de forma que tanto los lectores expertos como los inexpertos puedan evaluar la información correspondiente. En principio, todo lector desea conocer los resultados globales del experimento, preferiblemente de forma gráfica. Tal presentación puede realizarse con información cuantitativa más precisa, si bien los análisis numéricos detallados deben aparecer en forma de tablas.

Notas absolutas La presentación de las notas medias absolutas, para el objeto y la referencia oculta por separada, puede proporcionar una impresión inicial bastante acertada de los datos. La nota absoluta de la calidad MOS se suele denotar de la forma $\overline{MOS}$. Sin embargo, debe tenerse en cuenta que esto no constituye una base adecuada para realizar un análisis estadístico detallado, debido al hecho de que cuando se utiliza el método MOS los participantes saben explícitamente que una de las fuentes en la comparación por pares es idéntica a la referencia. En consecuencia, las observaciones no son independientes y el análisis estadístico de estas notas absolutas no aporta una información significativa.

Notas distintas La diferencia entre las notas otorgadas a la referencia oculta y al objeto es el punto de partida adecuado para efectuar los análisis estadísticos. Una representación gráfica revela claramente las distancias reales a la transparencia, que normalmente tienen gran interés. En este caso la nomenclatura usada para la diferencia entre las notas es $\Delta\overline{MOS}$.

Nivel de significación e intervalo de confianza El informe de la prueba debe explicitar los niveles de significación, así como otros detalles acerca de los métodos y resultados estadísticos que contribuyan a dar una idea más clara al lector. Dichos detalles podrían incluir los intervalos de confianza o las barras de error en los gráficos. Tradicionalmente, se elige el valor de 0.05 como nivel de significación.

Contenido de los informes de prueba

Los informes de prueba deben indicar, de la manera más clara posible, los métodos utilizados y las conclusiones extraídas. Deben presentarse detalles suficientes como para que, en principio, una persona con ciertos conocimientos pueda repetir el estudio a fin de verificar de forma empírica los resultados. Un lector informado debe ser capaz de entender e interpretar los detalles más importantes de la prueba, las razones fundamentales para el estudio, los métodos de diseño y ejecución del experimento y los análisis y conclusiones.

Debe prestarse atención a los puntos siguientes:

- Especificación y selección de los participantes y pasajes.
- Detalles físicos de los equipos y del entorno de escucha.
- Diseño del experimento, que incluye el adiestramiento, las instrucciones, las secuencias y procedimientos de prueba y la generación de datos.
- El procesamiento de los datos, incluyendo los detalles de los resultados estadísticos obtenidos.

- Conclusiones extraídas

2.7.2. El método MUSHRA

El problema de la escala MOS es que no ofrece un valor demasiado representativo para comparar varios codificadores, ya que no se pueden medir varios codificadores a la vez, sino que hay que realizar pruebas independientes entre ellos. Este problema se agrava cuando los codificadores de audio a evaluar introducen degradaciones medias, e incluso grandes, en la señal codificada. En estos casos, la dispersión de los resultados puede resultar inadmisible. Por ello ha surgido la recomendación [ITU-R01], donde se permite la inclusión de varios codificadores en la evaluación. Además, otra característica que introduce esta recomendación es la evaluación en cada test de audición de una señal fija de baja calidad, como es una señal paso bajo filtrada a 3,5 kHz. Esto permite tener una idea bastante aproximada de las desviaciones entre diferentes pruebas de audición. Así, cuando se trata de medir la calidad de codificadores por transfomada a bajo régimen binario o de algunos de los codificadores paramétricos donde la señal codificada no tiene ni mucho menos una alta calidad, se utiliza esta recomendación conocida como método MUSHRA. Además, es destacable reseñar que no existen alternativas fiables de medidas objetivas en esta situación. Así pues, el método comúnmente utilizado para medir la calidad de codificadores paramétricos es el método MUSHRA (*MUlti Stimulus test with Hidden Reference and Anchor*) [ITU-R01].

Para un tratamiento en más profundidad del método MUSHRA es conveniente acudir a [ITU-R01][Stoll00][Soulodre99], aunque se realiza a continuación una breve revisión de esta medida subjetiva. El método MUSHRA está basado en un test doblemente ciego y multi-estímulo, con una referencia oculta y una(s) señal(es) de prueba (*anchor(s)*), también oculta(s), especialmente diseñado para medir señales de audio con medias o grandes degradaciones de codificación. El test de audición se realiza en una o más sesiones. En cada sesión, el material de audio a calificar se presenta en varios intentos. En cada intento se presenta la misma señal de audio procesada de varias formas diferentes (o estímulos).

Un test es multi-estímulo cuando en un intento más de una forma de procesar la señal o estímulo se evalúa. El número de estímulos no debería en ningún caso exceder de 15. En el método MUSHRA la señal original sin codificar se usa como referencia. El material usado en el test incluye la referencia oculta, la(s) señal(es) de prueba y la(s) señal(es) codificada(s). La idea de proporcionar la referencia oculta es para poder asegurar la capacidad del oyente de detectar los artefactos de las señales codificadas y de prueba. El propósito de incluir señal(es) de prueba es para dar una comparación de la calidad del material codificado con respecto a niveles de calidad de audio bien conocidos. Así, al menos, se debe usar una señal de prueba, que normalmente es la señal original filtrada paso bajo. Normalmente, esta señal de prueba es la señal filtrada a 3,5 kHz, ya que este ancho de banda se utiliza con fines de supervisión en aplicaciones de difusión. En general, se pueden usar varias señales de prueba. Así, en la bibliografía relacionada con codificación paramétrica de audio se suele incluir como señal de prueba adicional una señal paso bajo a 7 kHz. La escala de medida usada en el método MUSHRA tiene cinco intervalos, aunque es una escala continua (*Continuous Quality Scale*, CQS), como se indica en la figura 2.18.

Cada intervalo de esta escala se corresponde con una puntuación de calidad que va desde 'mala' (correspondiente a una puntuación de 0 a 20) a 'excelente' (puntuación de 80 a 100).

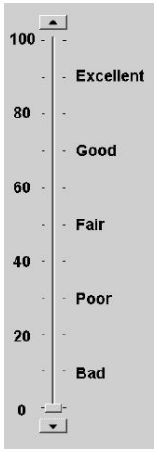

Figura 2.18: *Los cinco intervalos de la escala continua (CQS) de medida usada en el método MUSHRA.*

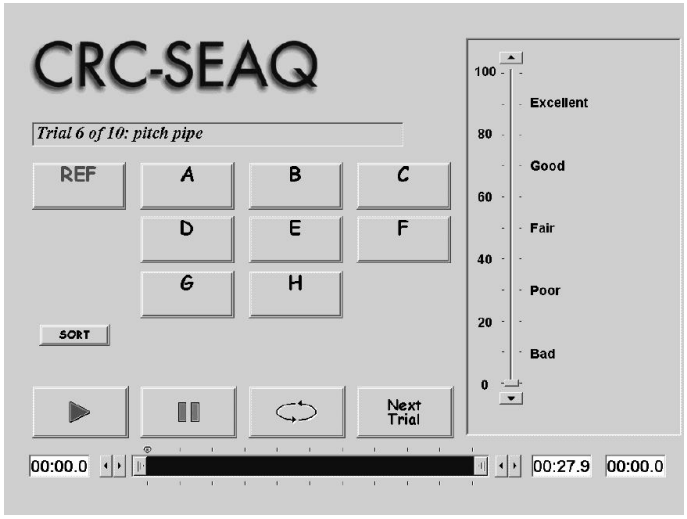

Figura 2.19: *El interfaz de usuario del programa SEAQ para realizar el test MUSHRA.*

Durante cada intento, el evaluador puede seleccionar cada estímulo en el orden que quiera. A continuación, debe puntuar la calidad de todos los estímulos de cada intento. El resultado de cada estímulo en todos los intentos y sesiones es el valor de calidad que proporciona el método MUSHRA, por lo que estará entre 0 y 100. Este procedimiento se puede realizar sin que una persona deba controlar el test. Así, en la figura 2.19 se incluye una ilustración de un programa usado para realizar el método MUSHRA. El programa conocido como SEAQ, y desarrollado por el centro de investigación de comunicaciones (CRC) de Ottawa, ofrece la posibilidad de elegir entre varios estímulos (de A a H en la figura), que incluyen la referencia, la(s) señal(es) de prueba y la(s) señal(es) codificada(s). En cada intento, las señales a evaluar son asignadas de forma aleatoria a cada estímulo. Para obtener unos resultados satisfactorios, los oyentes deben ser sujetos con conocimiento de los tipos de artefactos más críticos en las señales de audio. De hecho, el test debe ser precedido de una fase de entrenamiento donde cada evaluador se acostumbre a las señales de test y a los artefactos de codificación. Aunque ahora este procedimiento de selección y entrenamiento de oyentes no es tan crítico como en el test MOS, puesto que las señales son de peor calidad y, por tanto, las degradaciones son más fácilmente detectables cuando se presentan los resultados, los valores medios se deben acompañar de sus correspondientes intervalos de confianza que informen de la varianza de los resultados.

2.8. Conclusiones

En este primer capítulo se ha presentado un breve resumen del estado actual del arte en codificación perceptual de audio. Se han revisado los conceptos básicos necesarios para entender el funcionamiento de los sistemas reales y se han descrito las características más destacadas de los diferentes estándares ISO/MPEG audio. Además, puesto que en esta tesis doctoral se propone un esquema de codificación basado en una descomposición paramétrica del audio, en la parte de revisión de conocimientos se va a incluir un capítulo completo para revisar el estado del arte en relación a la codificación paramétrica de audio. En este capítulo se explicarán las técnicas de modelado de tonos, transitorios y ruido, así como las propuestas de codificadores con herramientas paramétricas encontradas en la bibliografía.

Para concluir este tema es preciso indicar que la codificación perceptual de audio es un tema aún candente debido a la necesidad de nuevas aplicaciones de codificación, como son la codificación escalable por internet o la codificación a bajo régimen binario por redes de telefonía móvil. Además, las herramientas que se desarrollen para conseguir la compresión de señal bajo estas circunstancias, al tratarse de modelos de señal paramétricos, se pueden utilizar en un amplio conjunto de aplicaciones diferentes. En este sentido, los parámetros de los modelos se pueden utilizar en un futuro para la clasificación de audio en base a contenido, o incluso, para la separación de fuentes.

Capítulo 3

Codificación paramétrica de audio

Los codificadores de audio por transformada, como MP3 [MPEG92], están diseñados generalmente para operar a múltiples regímenes binarios. En el caso de querer obtener un bajo régimen binario, se limita el ancho de banda de la señal de entrada con el objetivo de obtener una calidad satisfactoria en bajas frecuencias. Como consecuencia, el principal inconveniente de los codificadores por transformada es la rápida degradación de la calidad del audio cuando el régimen binario está por debajo de 40 kbits/s. Sin embargo, los codificadores de audio paramétricos, los cuales utilizan modelos de señal combinados con un modelo perceptual, son capaces de obtener una señal de audio codificada de alta calidad por debajo de 40 kbits/s. Además, la obtención de una representación paramétrica de la señal de audio permite realizar, fácilmente y de manera directa, modificaciones de la señal en el decodificador, tales como cambio de pitch y de escala temporal (tempo o *stretching*). En contraste con los codificadores de forma de onda, los codificadores paramétricos no aplican reducción de ancho de banda para reducir el régimen binario, sino que suelen ordenar los parámetros obtenidos de la señal según su importancia perceptual para conseguir escalabilidad en régimen binario [Verma99] [Myburg04].

El primer codificador completamente paramétrico aceptado en el estándar MPEG-4 es el conocido como HILN (Harmonic and Individual Lines plus Noise) [Purnhagen00] [MPEG99]. El codificador HILN puede operar en un rango de regímenes binarios que oscila de 6 a 16 kbits/s en mono. Aunque pobre, la calidad de audio conseguida por el HILN a esos regímenes binarios es comparable a la calidad obtenida por los mejores codificadores por transformada: TwinVQ [Iwakami95] a 6 kbits/s y AAC [MPEG97a] a 16 kbits/s, ambos en mono [Purnhagen00]. Más recientemente, el codificador paramétrico de audio desarrollado por Phillips [Brinker02] ha sido la respuesta de la empresa a la llamada a propuestas hecha en 2001 por MPEG [MPEG01]. Este codificador, conocido como PPC (Philips Parametric Coder), opera a un régimen binario de 24 kbits/s en estéreo, dando como resultado una señal de audio de mayor calidad que AAC a 24 kbits/s estéreo para la mayoría de las señales de prueba, salvo para las señales con transitorios y las señales vocales [Brinker02]. En Diciembre de 2003, MPEG anunció [MPEG03] que en la extensión 2 de MPEG-4 se incluía un codificador paramétrico de audio de alta calidad que coincide a grandes rasgos con el codificador PPC propuesto por Philips. Este codificador está diseñado para trabajar en el rango de 16 a 24 kbits/s por canal. Además, el decodificador permite el cambio de pitch y de tempo (*stretching*) en tiempo real [Kerkhof02] [Schuijers03]. Este resultado ilustra el potencial que puede llegar a tener la codificación paramétrica de audio, en el sentido

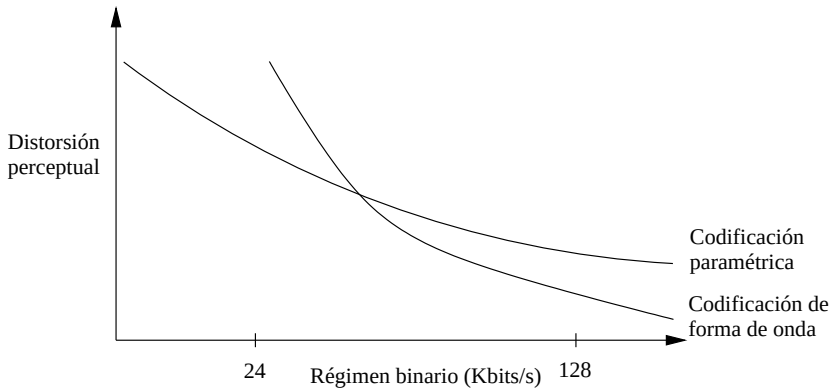

Figura 3.1: *Tendencia de la distorsión perceptual en función del régimen binario para codificadores de forma de onda y paramétricos.*

de que puede ser la herramienta óptima para la codificación de audio a bajo régimen binario. Sin embargo, el principal inconveniente radica en que, tanto para HILN como en los demás codificadores paramétricos, un incremento en régimen binario no se traduce en el consiguiente incremento en calidad de audio. La figura 3.1 compara la calidad de audio obtenida en función del régimen binario tanto para codificación paramétrica como para codificación de forma de onda. Como se puede observar, la codificación paramétrica supera en calidad a la codificación de forma de onda a bajo régimen binario. Sin embargo, si se quiere conseguir calidad de audio trasparente, con un alto régimen binario, hay que usar codificadores de forma de onda. Este hecho es el responsable de la combinación de HILN con codificación de forma de onda a alto régimen binario, resultando un codificador híbrido [Edler98]. Otro inconveniente de los codificadores paramétricos se produce en la codificación de voz, donde la calidad conseguida por los codificadores de voz para el mismo régimen binario es mayor. Así, con la intención de mejorar esta situación se ha usado el codificador HILN en combinación con codificadores de voz paramétricos (vocoders) [Edler98].

Un codificador paramétrico de audio necesita utilizar modelos de señal que extraigan las características de las diferentes componentes que conforman en la señal de audio. La señal de audio se descompone, de forma general, en tres componentes:

Tonos La componente tonal modela los elementos casi-estacionarios de la señal de audio. De las tres componentes de la señal de audio, los tonos es la componente dominante, porque requiere un mayor régimen binario, además de tener una fuerte contribución en la calidad final de la señal codificada. Los tonos se identifican como picos en la amplitud de la transformada de Fourier siendo, por tanto, sinusoides de lenta variación en el tiempo, conocidas en la literatura como parciales (del inglés *partials*). Una sinusoide (o tono) se parametriza mediante su amplitud, fase y frecuencia. La componente sinusoidal de la señal de audio son un conjunto de tonos con sus respectivos parámetros. Estos tonos pueden estar armónicamente relacionados en frecuencia, siendo entonces múltiplos de una frecuencia fundamental (o *pitch*). La componente sinusoidal del audio se debe principalmente a voz sonora y a música instrumental.

Los primeros indicios de modelado tonal en la literatura aparecieron asociados a los codificadores de voz que dividen la señal en una parte determinística y otra estocástica. El modelo determinístico es, básicamente, un modelo tonal como aparece en el vocoder de fase [Flanagan66]. Sin embargo, quizás el esquema tonal más usado posteriormente para codificadores de audio es el modelo presentado en [Mcaulay86], ideado en un primer momento sólo para señal de voz. En este modelo los tonos se detectan y extraen en cada trama de la señal de audio, por lo que antes es necesario segmentar la señal. Una vez que los tonos son detectados y extraídos de la señal, se puede conseguir una ganancia de codificación importante, agrupando los tonos entre tramas adyacentes para formar trayectorias tonales que duren más allá de una trama, llamadas caminos tonales (del inglés *tracks*). Cada camino tonal se caracteriza por la variación en el tiempo de la amplitud, frecuencia y fase del tono modelizado. Otro modelo para señal de voz que incluye también una señal estocástica es el vocoder de excitación multi-banda (Multi-Band Excitation, MBE) [Griffin88]. En el caso de la señal de audio, el modelo determinista de [Mcaulay86] se ha utilizado completándolo con un modelo estocástico en [Serra89]. Sin embargo, algunas señales de audio contienen ataques (por ejemplo, el sonido producido por una castañuela), que no puede representarse con un modelo tan sencillo.

Posteriormente, y centrándose en la señal de audio, se han utilizado diferentes estrategias para identificar y obtener los tonos. Un enfoque usado con asinuidad es el empleo del algoritmo *matching pursuits* [Mallat93], que extrae en cada iteración el tono más correlado con la señal. Este método se ha extendido, incluso para que tenga en cuenta información perceptual en el cálculo de la correlación, de forma que extraiga en cada iteración el tono más importante perceptualmente [Heusdens02]. Otro enfoque aplicado [Myburg04] se centra en la reducción del coste computacional, determinando de una vez todos los tonos, como en [Mcaulay86], evitando así un algoritmo iterativo. Por otro lado, se ha explotado la redundancia de la información de cada camino tonal aplicando codificación diferencial para las amplitudes y frecuencias de los tonos, explotando de esta forma su característica casi-estacionaria [Purnhagen00] [Brinker02] [Levine98] [Verma99]. En la práctica, la fase no se suele enviar en un codificador paramétrico, en su lugar el decodificador estima la fase para que la onda de la señal decodificada sea continua. Sin embargo, este esquema conduce a una señal decodificada poco natural [Levine98].

Transitorios La componente transitoria se refiere a los eventos no estacionarios de la señal de audio que se presentan típicamente en breves periodos de tiempo. La envolvente de un transitorio se caracteriza normalmente por un rápido incremento de la energía de la señal seguido de una caída de forma exponencial. Un buen modelo de parametrización de transitorios debe evitar que la energía del mismo se disperse y se produzca un pre-eco en la señal de audio decodificada. El sonido producido por un golpe de castañuela es un buen ejemplo de transitorio en la señal de audio.

La componente transitoria de la señal de audio debe ser tratada independientemente para conseguir una calidad de audio aceptable en las señales donde se presenta. Además, los transitorios son eventos poco frecuentes, por lo que el régimen binario necesario para su transmisión es bastante bajo, ya que la codificación de transitorios se habilita sólo cuando se detecta un transitorio. Se han propuesto varios modelos para la codificación de transitorios

en la literatura. Tanto en [Levine98] como en [Ali95] se aplica la codificación por transformada cuando aparece un transitorio, si bien [Levine98] utiliza la transformada modificada de coseno y [Ali95] un análisis wavelet. El primer esquema que modela los transitorios con parámetros es el propuesto en [Verma99], donde se aprovecha la dualidad entre tiempo y frecuencia. El enfoque aplicado en los codificadores paramétricos de audio estandarizados (PPC y HILN) se reduce a determinar la envolvente del transitorio y estimar el número de tonos bajo ésta [Brinker02] [Edler96].

Ruido La señal ruidosa se obtiene a partir del residuo que resulta de restar la señal original a la suma de la componente sinusoidal más la transitoria. La necesidad de conseguir un régimen binario bajo no permite la codificación de la forma de onda del residuo que dejan el modelo tonal y transitorio. En su lugar, se incluye un modelo de ruido que captura las características esenciales del residuo. Esta señal residual tiene características estocásticas, por lo que se parametriza su envolvente tanto espectral como temporal y su potencia. En el decodificador, se genera un ruido sintético con la misma potencia que el residuo del codificador para, posteriormente, adaptar mediante filtrado su envolvente espectral y temporal.

Por lo tanto, el modelo de ruido acepta la suposición de que el ruido es un proceso estocástico [Purnhagen00] [Brinker02] [Levine98] [Verma99]. La resolución, tanto temporal como espectral, que necesita el modelo de ruido para conseguir una señal de alta calidad perceptual debe tener en cuenta el sistema auditivo humano. Los métodos usados en la bibliografía tienen dos vertientes. Por un lado, una implementación mediante bancos de filtros [Goodwin97], y por otro, una implementación paramétrica basada en predicción lineal [Purnhagen00] [Serra97] [Brinker00]. La contribución de la componente de ruido en la calidad total de audio es de gran importancia, sólo detrás de la componente tonal. Si bien, ambos modelos deben estar bien sintonizados para poder obtener una señal decodificada de alta calidad. A grandes regímenes binarios, cuando se quiere conseguir una señal codificada de audio de calidad trasparente, es necesario codificar la forma de onda del residuo implementando un codificador híbrido [Edler98].

En la literatura aparecen varios ejemplos donde se utilizan los modelos sinusoidal y de transitorios, así como un modelo de ruido para el residuo correspondiente, con el objetivo de realizar una codificación paramétrica del audio. Esta codificación se conoce con el nombre de codificación STN (del inglés *Sines, Transients plus Noise*). En [Levine98], se aplica un modelo sinusoidal seguido de un modelo de residuo en los segmentos estacionarios de señal, mientras que en los transitorios se aplica codificación por transformada. En el caso del codificador HILN, se implementa una etapa inicial de análisis para detectar los transitorios, con el fin de modelizar su envolvente y reducir el tamaño de la trama [Edler96]. En cualquier caso, se aplica posteriormente un modelo tonal seguido de uno estocástico. En [Ali95], primero se aplica un modelo sinusoidal y tras extraer esta componente de señal, se analiza la componente transitoria. Básicamente, se aplica una transformada wavelet. En el caso de que una banda determinada tenga o no características estocásticas, se modela como ruido o se codifica directamente su forma de onda, respectivamente. Alternativamente, tanto en [Verma00] como en el codificador PPC [Brinker02], se aplica primero el modelo de transitorios. La razón de este cambio hay que buscarla en que el modelado sinusoidal

extrae muchos tonos cuando se aplica sobre una señal no estacionaria, de forma que si se aplica primero el modelo de transitorios se soluciona este inconveniente. A continuación, se repasan algunas de las técnicas más utilizadas para modelizar cada una de las componentes de audio.

3.1. Modelado sinusoidal

El modelo sinusoidal clásico [Mcaulay86] representa la señal de audio $x[n]$ como la suma de un conjunto de K sinusoides con frecuencias, fases y amplitudes variantes en el tiempo:

$$x[n] \approx \hat{x}[n] = \sum_{k=1}^{K} A_k[n] \cdot cos\left(\omega_k[n] \cdot n + \phi_k[n]\right) \tag{3.1}$$

donde $A_k[n]$, $\omega_k[n]$ y $\phi_k[n]$ representan la amplitud, frecuencia y fase del k-ésimo tono, respectivamente. En general, el comportamiento dinámico de la señal de audio se modeliza de forma correcta reconstruyendo la señal a partir de estos parámetros, asumiendo incluso que la amplitud $A_k[n]$ y frecuencia $\omega_k[n]$ del parcial k varía lentamente a lo largo del tiempo. Si se limita esta variación, la señal de audio se analiza segmentando la señal en tramas donde las amplitudes y frecuencias de los tonos son constantes. Por lo tanto, dentro de una trama (o segmento), la amplitud es constante A_k, y el argumento del coseno en la ecuación 3.1 es un polinomio lineal $\omega_k \cdot n + \phi_k$.

Para obtener los parámetros descritos que representan la componente tonal de la señal, el mecanismo más directo se basa en la identificación de picos espectrales en el espectro de amplitud de una trama enventanada obtenido mediante la transformada discreta de Fourier (Discrete Fourier Transform, DFT). La muestra de la DFT donde el espectro de amplitud es un máximo local proporciona una estimación de la frecuencia de un tono presente en la señal, y su valor complejo de la amplitud y fase del mismo. Sin embargo, este esquema tan sencillo debe ser completado con algoritmos que permitan discriminar si un máximo local coincide o no con un tono importante de la señal. Estos algoritmos suelen tomar como información la forma del espectro cerca del máximo, o bien, la predictibilidad (lo contrario de imprecibilidad) en frecuencia de la señal a lo largo del tiempo. Por ejemplo, la definición espectral en [Serra89] se basa en el nivel en dB del pico espectral en relación a las muestras cercanas de la DFT, con el objetivo de comprobar si coinciden con la transformada de la ventana utilizada. Sin embargo, siguiendo esta definición se pueden extraer tonos no-estacionarios que no formen trayectorias, sino que representen características aisladas de la señal. Para solucionar este inconveniente, el cálculo de la predictibilidad de la frecuencia del tono, mediante la señal en tramas adyacentes, mejora la estimación. En cualquier caso, como se verá posteriormente, se han presentado en la literatura un número amplio de algoritmos para mejorar la extracción tonal.

Los parámetros de cada tono (amplitud, fase y frecuencia) deben ser codificados para la compresión de la señal de audio. En este sentido, se consigue una ganancia de codificación significativa si se aplica un esquema de codificación diferencial para las amplitudes y frecuencias de los tonos entre las diferentes tramas. La explicación de este hecho hay que buscarla en que muchos tonos se repiten de un segmento al siguiente, debido a que describen la componente estacionaria de la señal de audio. Como consecuencia, un método de agrupamiento de tonos para conseguir una codificación diferencial inter-trama (o codificación diferencial en el tiempo) se presentó primera-

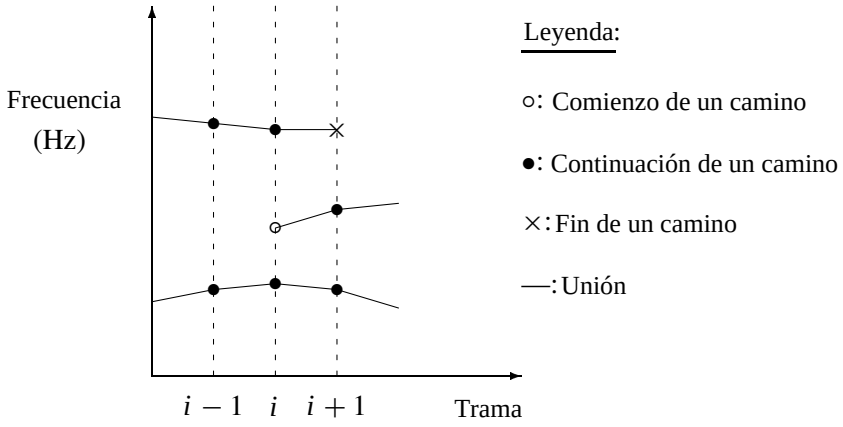

Figura 3.2: *Unión de tonos individuales para formar trayectorias*

mente en [Mcaulay86], y se ha aplicado en muchos de los codificadores presentados posteriormente [Ali95] [Brinker02] [Verma99] [Levine98] [Serra89]. Este método relaciona los tonos cercanos en frecuencia y amplitud entre tramas adyacentes para formar trayectorias (o caminos) tonales que se codifican de forma diferencial. Cuando aparece un tono en una trama, que no puede ser relacionado con un tono de la trama anterior, se comienza una nueva trayectoria. Cuando un tono se relaciona con otro de la trama anterior, se sigue una trayectoria antes comenzada. Si se puede relacionar un tono con otro de la trama siguiente se continúa la trayectoria; en caso contrario, la trayectoria se finaliza. En la figura 3.2 se presenta una gráfica tiempo-frecuencia que ilustra el comienzo, continuación o término de una serie de trayectorias tonales.

Adicionalmente, se puede conseguir una reducción del régimen binario si no se codifican las fases de los tonos. En su lugar, el decodificador aplica un algoritmo de continuación de fase que evita que haya discontinuidades de señal en las fronteras entre tramas. Sin embargo, este esquema lleva consigo la reproducción de una señal de audio poco natural, especialmente para señales de voz [Brinker02]. Otra alternativa para disminuir la cantidad de datos en el proceso de codificación, aparecida en [Jensen02], es aplicar técnicas basadas en codificación diferencial intra-trama, relacionando los tonos dentro de una misma trama.

3.1.1. Psicoacústica aplicada al modelo tonal

Para realizar un diseño apropiado de los parámetros del modelo sinusoidal, es necesario realizar una revisión del modelo auditivo del oído humano. Por un lado, es imprescindible tener en cuenta al segmentar la señal de audio la resolución temporal que tiene el oído. Por otro, es necesario parametrizar las frecuencias de los tonos con la aproximación suficiente, adaptada a la percepción humana, para conseguir una buena calidad del modelo. Incluso, es interesante realizar una extracción tonal guiada perceptualmente, hasta el punto de conseguir extraer todas las sinusoides percibidas por el oído, siendo necesario para este propósito el cálculo del umbral

de enmascaramiento sobre la componente tonal de la señal.

Resolución temporal y espectral

La resolución espectral del modelo tonal es inversamente proporcional a su resolución temporal. Así, la capacidad de discriminar entre dos frecuencias del modelo, o resolución espectral, crece conforme la duración de la trama de análisis es mayor, y se reduce la resolución en el tiempo. Por lo tanto, el tamaño de trama de análisis debe ser lo suficientemente grande como para poder diferenciar dos frecuencias independientes de la señal de audio. En general, este no es un problema crítico, ya que las señales tonales son, frecuentemente, múltiplos de una frecuencia fundamental o *pitch*, cuyo rango es conocido. Bien es verdad que los tamaños de trama comúnmente utilizados pueden provocar la interferencia entre tonos debida a los lóbulos laterales de la ventana de análisis. Sin embargo, el uso de ventanas de Hanning o Hamming [Harris78] solventa este problema.

En cuanto a la resolución temporal, ésta tiene que ser suficiente para modelizar correctamente los tonos de alta frecuencia cuyas características varían de forma rápida en el tiempo. Al final, la elección del tamaño de trama se realiza aplicando un valor de compromiso para poder tener una buena discriminación en frecuencia y tiempo. La figura 3.3 ilustra el intercambio entre resolución temporal y espectral cuando se considera un fragmento de voz sonora masculina. En esta figura se representa la señal en el tiempo para tres longitudes de trama diferentes, todas centradas en $t = 0$. La duración de cada trama es de (a) $100ms$, (b) $40ms$ y (c) $10ms$. El espectro de amplitud en dB aplicando la DFT con ventana de Hanning, correspondiente a cada trama, se dibuja en los apartados (d), (e) y (f), respectivamente. Como se puede observar en el dibujo del apartado (d), todas las frecuencias están armónicamente relacionadas, si bien por debajo de $1{,}2kHz$ están bien definidas, mientras que en frecuencias más altas no sucede lo mismo, debido a que los cambios de estas últimas en el tiempo se producen de forma más rápida que el tamaño de trama usado. En el apartado (e) se definen correctamente todos los picos espectrales en el rango presentado, por lo que se puede afirmar que es un buen valor de compromiso. El apartado (f) presenta el resultado de seguir disminuyendo el tamaño de la trama de análisis; como era de esperar, se ha reducido tanto el tamaño de trama que ya no se pueden distinguir los tonos individuales presentes en la señal, pues la resolución espectral es insuficiente.

El empleo del relleno con ceros (*zero padding*) para el cálculo de la DFT no soluciona el problema de la discriminación de dos tonos muy juntos en frecuencia, debido a que este mecanismo simplemente interpola en frecuencia el espectro, dándole un aspecto más suave. Sin embargo, tiene como ventaja que permite una detección más aproximada de la estimación de la frecuencia. En la práctica, la exactitud de la obtención del parámetro de frecuencia de cada tono es importante, porque el oído humano tiene gran discriminación en baja frecuencia [Zwicker90], lo que lleva al uso generalizado del relleno con ceros. Además, este mecanismo se debe utilizar para calcular la DFT mediante el algoritmo FFT (Fast Fourier Transform) con longitudes de entrada que sean potencias de 2.

La habilidad del oído humano para discriminar mejor las bajas frecuencias está relacionada con el funcionamiento interno del mismo, lo cual debe ser tenido en cuenta a la hora de determinar la discriminación del modelo tonal a lo largo de la frecuencia. Desde un punto de vista perceptual, el sistema auditivo humano tiene una primera etapa de pre-procesamiento, que se puede modelar

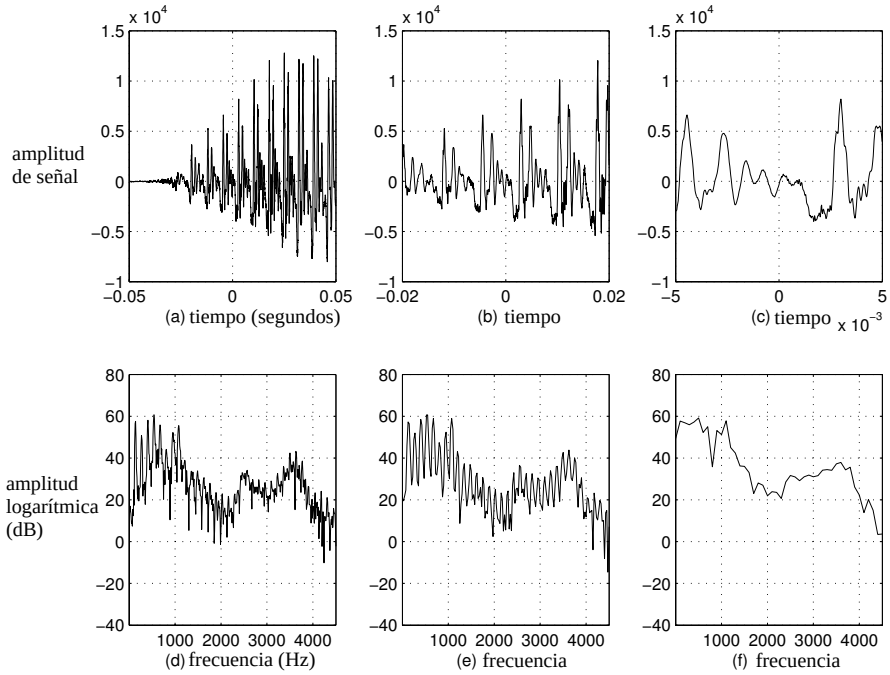

Figura 3.3: *Evolución de la resolución espectral y temporal con el tamaño de trama de análisis. Las señales en el tiempo se presentan en los apartados (a), (b) y (c), donde se reduce la duración de la trama. Sus correspondientes espectros de amplitud en dB mediante la DFT de la señal enventanada con la ventana de Hanning se presenta en los apartados (d), (e) y (f), respectivamente.*

por un banco de filtros paso banda. Este banco de filtros tiene la característica propia de tener anchos de banda pequeños en baja frecuencia, que se incrementan en frecuencia. Los anchos de banda de este banco de filtros auditivo se aproximan con la escala de Bark (de Barkhausen) [Zwicker90], o escalas similares de carácter logarítmico [Glasberg90], y se conocen como bandas críticas. En el oído interno se compone la señal de salida de las bandas críticas, por lo que si dos tonos están en la misma banda crítica el oído humano tendrá menor capacidad de distinguirlos que si están en bandas críticas diferentes. Como conclusión, la discriminación en frecuencia del oído es mucho mayor en baja frecuencia y depende de la escala de Bark.

Como cualquier analizador tiempo-frecuencia, el oído tiene una mayor discriminación temporal donde su discriminación espectral es menor. Como consecuencia, la resolución temporal del oído es mucho mejor en alta frecuencia [Zwicker90]. Este hecho sugiere la necesidad del empleo de un análisis con múltiples escalas temporales en función de la frecuencia para adaptarse a las características del oído humano, análisis que se conoce como análisis multi-resolución. En la práctica, se suelen utilizar tramas de larga duración en el tiempo para las bajas frecuencias y de corta duración para las altas frecuencias en la mayoría de los codificadores paramétricos de audio [Brinker02] [Levine98] [Verma99] [Goodwin97] (aproximando la resolución que se obtiene con un banco de filtros wavelet). En la mayoría de los casos, sólo se utilizan dos o tres escalas diferentes por simplicidad. Las ventajas del análisis multi-resolución se aprecian en la figura 3.4, donde se analiza el sonido de una gaita. En esta figura, se presenta la señal en el tiempo, centrada en $t = 0$, con diferentes tamaños de trama: (a) $10ms$ y (b) $80ms$. El espectro de amplitud en dB, aplicando la DFT con ventana de Hanning, se presenta respectivamente en las gráficas (c) y (d). La frecuencia fundamental de la componente tonal se sitúa aproximadamente en $640Hz$, pero sólo se observa de forma clara en la gráfica (d), donde el rango de frecuencias se sitúa de 0 a $2kHz$. En la gráfica (c), la corta duración de la trama temporal no permite una visión clara de este valor. Un estudio multi-resolución de la señal permite, por lo tanto, un compromiso adaptado de resolución tiempo-frecuencia para todo el rango de frecuencias.

Pese a que en la mayoría de los codificadores se realiza un análisis multi-resolución, las longitudes de trama para cada escala son en general fijas, sin tener en cuenta ningún conocimiento sobre el comportamiento local de la señal. Así, la longitud de los segmentos de señal se elige como un compromiso entre la variabilidad de la señal y la limitación en régimen binario. La primera alternativa a este problema consiste en realizar una segmentación en función de la frecuencia fundamental del complejo armónico de la señal [Kleijn95] [Serra97], como se hace en voz. Este esquema segmenta la señal de forma que se agrupan unos pocos periodos (tres o cuatro típicamente) de su frecuencia fundamental, con el fin de tener una resolución espectral suficiente para discriminar entre tonos armónicos y adaptarse a las variaciones de señal. Es posible agrupar mayor longitud de señal si se adapta a los cambios de ésta. Por ejemplo, cuando aparece un crecimiento de la frecuencia fundamental en el tiempo (efecto *chirp*), se puede transformar la señal en el eje temporal para que el periodo sea constante [Sluijter99]. Los inconvenientes que tiene esta técnica, muy útil en señal de voz, son: hay que detectar la frecuencia fundamental de forma exacta (lo que añade complejidad) y la señal de audio puede no tener frecuencia fundamental (cuando el conjunto de tonos no tiene relación armónica).

Sin embargo, se puede admitir revisando la bibliografía que una alternativa muy interesante al análisis multi-resolución es la segmentación adaptativa, que aplica una longitud variable de la trama de análisis [Xiong97] [Prandoni97]. La ventaja de este mecanismo radica en la posi-

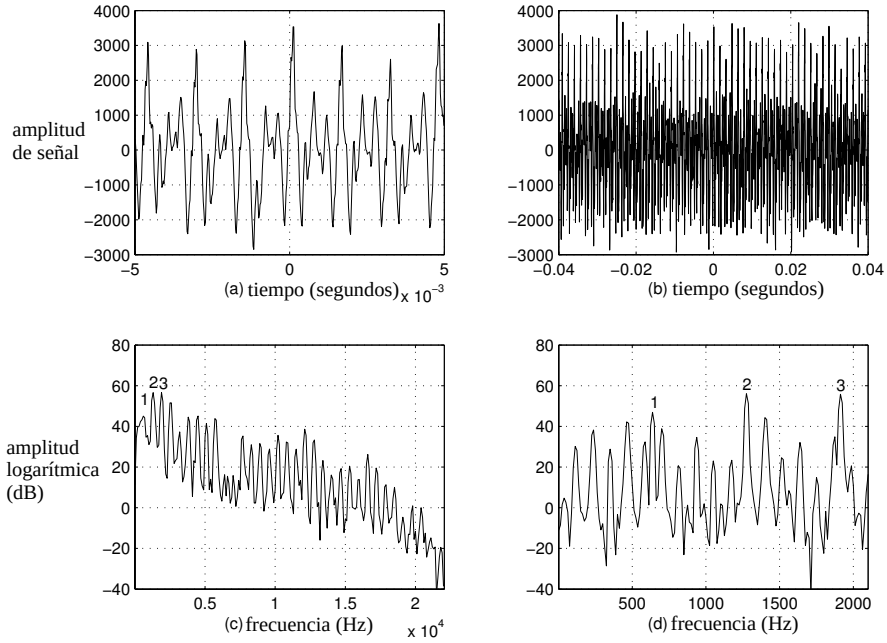

Figura 3.4: *Ventajas del análisis multi-resolución. Una trama, centrada en* $t = 0$*, con* $10ms$ *de duración se presenta en el apartado (a), mientras que su espectro de amplitud en dB se dibuja en el apartado (c). Una trama más larga de* $80ms$*, centrada también en* $t = 0$*, aparece en el apartado (b), con su espectro en dB en (d). Se compara la discriminación de los tres primeros armónicos en las gráficas (c) y (d).*

bilidad de adaptar la longitud de la trama al comportamiento local de la señal. Así, las partes estacionarias de la señal de audio se deben modelar con tramas mayores, mientras que en las partes no estacionarias las fronteras de las tramas se tienen que adaptar para evitar artefactos del tipo pre-eco de la señal codificada. Sin embargo, el precio a pagar es el incremento de la complejidad computacional. Mediante el uso de segmentación adaptativa, en [Painter01] se observa que se pueden alcanzar mejores resultados en régimen binario que con análisis multi-resolución con tramas fijas para cada escala. Es más, se puede incluso conseguir una distorsión mínima del modelo sinusoidal aplicando este esquema [Heusdens02b]. Como se ha observado, el principal inconveniente de la segmentación adaptativa es su alta complejidad computacional, especialmente si se quiere obtener un tamaño de segmento óptimo. Ante la necesidad de algoritmos de baja complejidad para conseguir segmentación adaptativa, han aparecido en la literatura las varias propuestas [Gonzalez01] [Ruiz02].

Umbral de enmascaramiento para la componente tonal

En codificación paramétrica de audio, el bajo régimen binario deseable restringe generalmente el número de tonos que se pueden codificar en el modelo sinusoidal. Para obtener una señal de audio decodificada de alta calidad, es necesario seleccionar el conjunto de tonos a transmitir de forma que la distorsión perceptual producida sea lo menor posible. En este sentido, si la distorsión perceptual está por debajo del umbral de enmascaramiento, se considera que ésta es inaudible, siendo la señal decodificada una copia trasparente, desde el punto de vista perceptual, de la señal original. Sin embargo, debido al bajo régimen binario, este resultado es demasiado

ambicioso en los codificadores paramétricos, por lo que se reduce a conseguir la mejor calidad perceptual posible.

En el cálculo de la distorsión perceptual de un tono determinado se tiene en cuenta el umbral de enmascaramiento de la componente tonal. A continuación, se presenta un resumen de las características principales del umbral de enmascaramiento para la componente tonal. Una descripción detallada de modelos de enmascaramiento se puede encontrar en [Zwicker90], [MPEG92], [Painter00] o [Par02]. Hasta la aparición de los codificadores paramétricos de audio, el umbral de enmascaramiento calculado era el de ruido, puesto que la distorsión producida en el proceso de codificación era el ruido de cuantificación de alguna transformada de la forma de onda. Sin embargo, hace tiempo que se conocen las propiedades de enmascaramiento entre tonos y entre ruido y tonos [Zwicker90]. En general, el cálculo del umbral de enmascaramiento para los tonos sigue el mismo algoritmo que para el caso del ruido, aunque tiene características propias. Este umbral se calcula analizando segmentos de señal enventanada en el dominio de la frecuencia. Al igual que en el caso del ruido, hay que clasificar la señal de entrada en máscaras tonales y ruidosas, puesto que los tonos son máscaras menos efectivas que el ruido [Hawkins50] [Zwicker82]. Para realizar esta clasificación, se puede analizar la forma del espectro de la señal en cada banda crítica [Johnston88b], o bien calcular la predictibilidad de cada pico espectral de la transformada [MPEG92]. Sin embargo, en el caso de un codificador paramétrico, esta distinción es directa si se aplica antes la extracción tonal. Tras esta separación, se evalúa el enmascaramiento que produce cada máscara mediante la función de dispersión (*spreading* en la bibliografía en inglés). Esta función de dispersión se utiliza para simular el efecto del banco de filtros paso banda del sistema auditivo humano y expresa la potencia que produce la máscara a la salida de este banco de filtros. El umbral de enmascaramiento final se calcula sumando las potencias de cada máscara tras la función de dispersión [MPEG92].

Una vez obtenida la máscara, ésta se utiliza para evaluar la distorsión perceptual producida por el proceso de codificación. Ahora bien, para realizar este cálculo correctamente, hay que tener en cuenta algunos principios psicoacústicos. En primer lugar, para deteminar si una distorsión es audible, hay que compararla con el umbral de enmascaramiento en toda la frecuencia, puesto que el oído integra todas las distorsiones del banco de filtros auditivos [Langhans92] [Buus86]. Además, el sistema auditivo humano es capaz de integrar la información acústica a lo largo del tiempo, más allá de la duración de una trama. Este tiempo se estima en unos $300ms$ [Brink64]. Por lo tanto, la detectabilidad de una distorsión aumenta con su duración, lo que debe ser tenido en cuenta por el modelo psicoacústico [Par02].

En codificación paramétrica de audio, la importancia perceptual de un pico espectral se evalúa mediante la relación señal-a-máscara (Signal-to-Mask Ratio, SMR), que es la distancia entre la potencia del pico espectral y la máscara a esa frecuencia [Verma99b]. Sin embargo, esta definición no tiene en cuenta ni la integración de la frecuencia tras el banco de filtros auditivos, ni la integración temporal, más aún cuando un tono representa la parte estacionaria de la señal de audio. Ambas cuestiones se han tenido en cuenta en algunos modelos tonales propuestos. Así en [Heusdens02], se integra sobre toda la frecuencia para calcular la importancia perceptual. En [Levine98], se da mayor importancia perceptual a aquellas trayectorias tonales de mayor duración. En cualquier caso, una vez calculada la importancia perceptual de cada tono mediante el umbral de enmascaramiento de la componente tonal, esta información no sólo puede servir para determinar los tonos audibles, sino que se debe tener en cuenta para clasificar los tonos

desde un punto de vista perceptual.

3.1.2. Tonos con relación armónica y tonos aislados

La componente sinusoidal de la señal de audio se divide en dos sub-componentes: una armónica y otra de tonos aislados. La sub-componente armónica se incluye por dos razones. La primera es que los sonidos armónicos ocurren de forma natural en la señal de audio, unas veces aislados y otras en combinación con otros sonidos. La segunda razón hay que buscarla en la reducción de información que supone la representación de la señal en: frecuencia fundamental y número de armónicos. Esta representación es muy eficiente sobre todo cuando el número de armónicos es alto. Los tonos aislados se tienen que incluir para modelar los tonos que no tienen relación armónica. Esta clasificación de los tonos aparece en algunos codificadores de audio [Serra97] [Purnhagen98] [Masri96]. Sin embargo, en otros, como el codificador PPC [Brinker02], dicha clasificación no se tiene en cuenta, puesto que en la señal de audio puede haber más de un complejo armónico y esto complica sobremanera el codificador.

En teoría, las frecuencias con relación armónica son múltiplos enteros de la frecuencia fundamental ω_f. El modelo más simple de un complejo armónico es aquel donde las amplitudes y frecuencias de los tonos son constantes durante la duración de una trama:

$$s_{hc}[n] = \sum_{k=1}^{N_h} A_k \cdot cos\left(k \cdot \omega_f \cdot n + \phi_k\right) \tag{3.2}$$

En esta expresión, N_h es el número de armónicos, A_k la amplitud del armónico k-ésimo y ϕ_k su fase. Los algoritmos para la estimación de la frecuencia fundamental ω_f se han desarrollado primero en el campo del tratamiento de señal de voz, y son diversos [Kleijn95] [Hess83] [Rabiner78] [Sondhi68].

De forma similar, el modelo para tonos aislados con amplitudes y frecuencias constantes es

$$s_{is}[n] = \sum_{k=1}^{N_c} A_k \cdot cos\left(\omega_k \cdot n + \phi_k\right) \tag{3.3}$$

donde N_c es el número de tonos y ω_k la frecuencia del tono k.

Si bien este modelo tan básico se ha usado en algunos codificadores paramétricos de audio es generalmente inadecuado para un rango amplio de señales de audio [Myburg04].

Complejo armónico

El modelo matemático de la ecuación (3.2) para un complejo armónico es demasiado simple para su empleo en la práctica debido a dos razones: 1) los cambios no estacionarios que se producen en algunas señales armónicas y 2) las propiedades no lineales en frecuencia de algunos instrumentos.

Los cambios en el complejo armónico de algunas señales afectan a varios de sus parámetros, siendo la señal de voz la que tiene típicamente un comportamiento dinámico más cambiante. Como se puede observar en la figura 3.3, la variación de amplitud de la señal ha de tenerse en cuenta en la duración de una trama. Una solución consiste en el modelado de la amplitud de los tonos mediante un polinomio de orden bajo. Otro aspecto a tener en cuenta es el cambio de la

frecuencia fundamental con el tiempo, lo que afecta a todo el conjunto de armónicos. Un modelo para este efecto [Sluijter99] se basa en suponer una variación lineal de la frecuencia fundamental ($\omega_f + 2\xi_c n$), donde ξ_c es el cambio de frecuencia y n el tiempo discreto. La estimación del parámetro de cambio de frecuencia ξ_c se realiza aplicando modificaciones sobre el eje temporal (*time warping*) [Sluijter99].

En cuanto a las propiedades no lineales de algunos instrumentos de cuerda, como el piano, el efecto en términos de señal corresponde a que la relación entre la frecuencia fundamental ω_f y el resto de frecuencias armónicas no es un número entero, sobretodo para los armónicos de alta frecuencia. Este efecto, conocido como *inharmonicity* en la literatura, se puede predecir para el armónico k mediante:

$$k\omega_f\sqrt{1+Bk^2} \tag{3.4}$$

donde el parámetro B vale

$$B = \frac{\pi^3 d^4 E}{64TL^2} \tag{3.5}$$

y depende de las características físicas de la cuerda como: el diámetro d, el factor de Young o elasticidad del material E, su tensión T y su longitud L [Fletcher64]. Hay varias técnicas en la literatura para estimar el parámetro B. En [Lattard93], se obtiene aplicando la ecuación (3.4) a partir de las frecuencias estimadas. En [Galembo99], se realiza una búsqueda sobre un rango especificado de valores de B. Los mismos autores en [Askenfelt00] aplican técnicas basadas en el cepstrum.

Pese a este desarrollo de mecanismos para utilizar el complejo armónico, el codificador HILN [Purnhagen98] es el único que distingue esta sub-componente, teniendo en cuenta las propiedades de los instrumentos de cuerda. Así, la frecuencia del armónico k-ésimo se calcula como ([MPEG99b, anexo A])

$$k\omega_f(1+\tilde{B}k^2) \tag{3.6}$$

En el codificador HILN, primero se estima la frecuencia fundamental a partir del cepstrum de la señal. Después, se extraen todos los armónicos, para calcular el parámetro $\tilde{B}$ mediante un algoritmo iterativo [Edler96] que minimiza el error entre los tonos calculados mediante (3.6) y los previamente estimados [MPEG99b].

Tonos aislados

Como en el caso del complejo armónico, el modelo simplista propuesto para los tonos aislados en 3.3 es inadecuado cuando los tonos no son estacionarios. Este modelo se ha complementado para tener en cuenta la no estacionareidad tanto en amplitud como en frecuencia. Para la no estacionareidad en las amplitudes, se han llegado a proponer modelos de tonos que abarcan el modelado de transitorios. Así, en [Goodwin97] [Friedlander95] se hace referencia al modelado de la señal con amplitudes tonales exponenciales $A_k[n] = A_k e^{\gamma_k n}$, con $\gamma_k < 0$. Esta extensión del modelo tonal se utiliza en [Heusdens00] [Nieuwenhuijse98], en combinación con una segmentación adaptativa dependiente de la señal, para el modelado de transitorios en un codificador paramétrico de audio. También hay propuestas menos ambiciosas con amplitudes tonales

polinómicas [George87]. La no estacionareidad en frecuencia se modela generalmente mediante fase polinómica [Goodwin97]. Por ejemplo, en [George87] se considera la variación de fase como $\Psi_k[n] = \phi_k + \omega_k n + \xi_k n^2$ siendo suficiente para los tonos de la señal de audio.

3.1.3. Métodos para mejorar la extracción tonal

La obtención óptima de los parámetros de un tono (amplitud, fase y frecuencia) es un problema difícil, debido a que frecuencia y fase están contenidas dentro del argumento de la función coseno. Como se ha visto anteriormente, el empleo de la transformada de Fourier es la primera piedra de toque en el cálculo de estos parámetros. La distribución tiempo-frecuencia que desarrolla la transformada de Fourier para una señal segmentada se conoce como STFT (Short-Time Fourier Transform) [Cohen95]. Esta distribución es computacionalmente eficiente gracias al algoritmo FFT, aunque su resolución es poco flexible. Así, la descomposición se realiza con funciones base exponenciales complejas con amplitud constante y, por tanto, estacionarias en la trama de análisis. Si la señal de audio es no es estacionaria en un segmento (que para este caso es similar a decir que la amplitud o frecuencia de los tonos varíen en un segmento), la descomposición obtenida mediante la STFT es poco útil. Sin embargo, para la mayoría de los segmentos de audio, la estacionareidad es suficiente para la STFT. La asignación de tonos a los picos espectrales de la STFT es un algoritmo directo y simple de obtención de parámetros del modelo sinusoidal. Sin embargo, la exactitud del modelo, sobre todo en frecuencia, está limitada por el muestreo en frecuencia de la DFT y es insuficiente. Se necesitan algoritmos de obtención más robustos, especialmente desde un punto de vista psicoacústico, por lo que han aparecido en la literatura una serie de métodos con aplicación a codificación paramétrica de audio para mejorar la extracción tonal de la STFT. Entre estos métodos se destacan los siguientes:

Relleno con ceros e interpolación. Si se rellena con ceros la señal enventanada antes de realizar la DFT, el efecto en el espectro es la interpolación. Como resultado el máximo de los picos espectrales se puede obtener de una forma más aproximada, lo que redunda en una estimación de frecuencia más exacta. [Serra97].

Derivada de la señal. La aplicación de la derivada de la señal para obtener una frecuencia más aproximada se realiza en [Desainte00].

Análisis de la distorsión de fase. La forma de la fase del espectro en las zonas próximas a un pico espectral permite estimar si el tono tiene una amplitud exponencial. También es posible determinar si se produce un cambio de frecuencia del tono durante la trama de análisis [Masri96] [Masri98].

Optimización no lineal restringida. En [Hamdy99] se formula un problema de optimización no lineal con búsqueda limitada de soluciones. La función de coste a minimizar propuesta es similar a

$$c = \|(x[n] - s_{modelo}[n]) \cdot w[n]\|^2 \tag{3.7}$$

donde $x[n]$ es la señal original en la trama actual de tamaño N_s, $s_{modelo}[n]$ es la señal sintética obtenida mediante los parámetros del modelo tonal y $w[n]$ es la ventana utilizada.

La búsqueda de los parámetros tonales se restringe a un ámbito reducido. Por ejemplo, para el caso de la frecuencia, el rango de búsqueda es:

$$\omega_{k,inicial} - \frac{B_s}{2} < \omega_{k,mejorada} < \omega_{k,inicial} + \frac{B_s}{2} \tag{3.8}$$

donde $B_s = \frac{2\pi}{N_s}$ es la separación de las muestras del espectro de la DFT en rad/s, $\omega_{k,inicial}$ es la estimación inicial de la frecuencia, y $\omega_{k,mejorada}$ la estimación mejorada de la frecuencia. También se restringen las amplitudes de una manera similar. La soluciones aportadas por este planteamiento conducen a una detección considerablemente exacta de los parámetros de frecuencia y amplitud de cada tono.

Optimización por el método de Gauss-Newton. En [Depalle97] se utiliza el método numérico de Gauss-Newton para obtener una estimación más aproximada de frecuencia y fase. El modelo usado es el más sencillo, puesto que tanto frecuencia como amplitud son constantes en la trama de análisis. Los resultados obtenidos manifiestan una alta sensibilidad a la forma de la ventana, por lo que se necesita el uso de ventanas sin lóbulos laterales. En el proceso iterativo de optimización los tonos muy próximos en frecuencia se llegan a fusionar en uno solo.

Método de Newton. En [Vos99] se realiza una búsqueda de soluciones óptimas haciendo uso del gradiente o método de Newton para mejorar la detección inicial de la DFT. Este método se ha extendido, permitiendo amplitudes exponenciales de los tonos en [Heusdens00].

Análisis de la fase. El análisis de la fase de la transformada como un polinomio de la forma $\Psi_k[n] = \phi_k + \omega_k n + \xi_k n^2$ permite la detección bastante exacta de la frecuencia ω_k y del cambio de frecuencia ξ_k. La primera propuesta que aparece en la bibliografía [Tretter85] tiene el inconveniente de tener que obtener una función de fase lineal (*unwrapping*). Este inconveniente es evitado en [Kay89] mediante el uso de datos de fase diferenciales, por lo que esta técnica se ha incluido con éxito en codificadores paramétricos de audio [Edler96].

Algoritmo matching pursuits. Este es uno de los métodos más utilizados en codificación paramétrica de audio. El origen de este enfoque hay que buscarlo en el empleo de descomposiciones atómicas, que se tratará con detenimiento en el próximo capítulo. Básicamente, la idea es descomponer la señal enventanada en un conjunto de funciones que pertenecen a un diccionario. Este diccionario debe tener un conjunto amplio de funciones, es decir, debe ser sobre-completo. La ventaja que permite el empleo de un diccionario de estas características es la de descomponer la señal en sólo unas pocas funciones (con el objetivo de comprimir al máximo la señal). El algoritmo más utilizado para realizar la descomposición es *matching pursuits* [Mallat93]. Este es un algoritmo iterativo que en cada iteración elige, para el caso del modelo sinusoidal, el tono más correlado con la señal, es decir, aquel tono que extrae más energía de la señal. La principal ventaja del algoritmo *matching pursuits* radica en que no se extraen picos espectrales debidos a lóbulos laterales [Myburg04]. Sin embargo, como inconveniente, es destacable que si se produce una extracción errónea de un tono, esta decisión puede afectar a otros tonos extraídos posteriormente. Este algoritmo ha sido extendido incluyendo información psicoacústica con el objetivo de extraer en cada iteración el tono más importante perceptualmente [Verma99b] [Heusdens02b].

Estimación por máxima semejanza. Este enfoque ha surgido en la literatura para otras aplicaciones; en concreto, para usos geofísicos. En esta técnica se emplea un conjunto de ventanas ortogonales, llamadas ventanas discretas esferoidales, y usa además el test estadístico *F-test* para decidir si un tono existe en una frecuencia particular [Thomson82]. El test estadístico ayuda a la discriminación entre picos espectrales y lóbulos secundarios puesto que da valores altos en el primer caso.

Minimización por mínimos cuadrados. Esta técnica está diseñada expresamente para reducir la complejidad, con el objetivo de evitar una búsqueda iterativa demasiado costosa en ocasiones para la codificación en tiempo real. Por lo tanto, se extraen todos los tonos se forma simultánea, directamente a partir de los picos espectrales [Mcaulay86]. Posteriormente, en lugar de elegir los parámetros de amplitud y fase directamente del espectro, se resuelve un conjunto de ecuaciones lineales que minimizan por el método de mínimos cuadrados el error entre la señal original y la obtenida por el modelo [George87]. La ventaja de este enfoque es que no hay propagación del error. Sin embargo, se pueden identificar como tonos picos espectrales debidos a los lóbulos laterales de la ventana de análisis.

Otras distribuciones tiempo-frecuencia. Como se ha visto, el empleo de la STFT tiene una serie de dificultades, siendo la principal de ellas que no está pensada para que la frecuencia del tono cambie durante una trama. Por esta causa, han surgido en la literatura una serie de distribuciones adaptadas al carácter no estacionario de la señal y que ofrecen otras resoluciones tiempo-frecuencia. Cuando la señal de entrada tiene un único tono, la distribución mejor adaptada para estima un cambio de frecuencia lineal es la distribución de Wigner (WD) [Cohen95]. Ahora bien, cuando hay varios tonos en la señal, esta distribución no es viable porque surgen productos cruzados en el plano tiempo-frecuencia. Como solución a este problema se pueden emplear las clases de Cohen, las cuales proporcionan un conjunto de distribuciones tiempo-frecuencia [Cohen95]. El cálculo de la transformada de Fourier sobre un plano tiempo-frecuencia con una distribución de este tipo permite la extracción de los parámetros tonales. Esta transformada se conoce como la función de ambigüedad (AF) para la distribución de Wigner [Peleg91] o función de ambigüedad de orden elevado (HAF) para las distribuciones de Cohen. Aún calculando la función HAF, se producen algunos términos cruzados entre los tonos, por lo que en [Barbarossa98] se presenta una modificación llamada función ambigüedad de orden elevado mediante producto (PHAF) que suprime estos términos. El único problema residente con la función PHAF aparece cuando las amplitudes de los tonos tienen un rango dinámico amplio. Un algoritmo iterativo apropiado en codificación paramétrica de audio que hace uso de estas técnicas aparece en [Ikram01]. Este algoritmo, al igual que *matching pursuits*, extrae en cada iteración el tono con mayor peso, aunque en la función PHAF en este caso, por lo que tiene el mismo problema de propagación de error.

A la vista de la revisión de métodos realizada, se puede concluir que hay un gran número de opciones a la hora de elegir el método a utilizar en un codificador paramétrico de audio. Si bien los mejores resultados subjetivos se han conseguido con el algoritmo *matching pursuits* con adaptación psicoacústica [Heusdens02b], éste tiene una complejidad demasiado elevada para su

implementación en tiempo real. Las diferentes opciones elegidas por los codificadores paramétricos de audio propuestos en la bibliografía se comentarán posteriormente.

3.2. Modelado de transitorios

El modelado sinusoidal combinado con un modelado de ruido adecuado es un modelo simple y eficiente para un conjunto amplio de señales de audio. Sin embargo, este modelo mixto de dos componentes no puede modelizar de forma eficiente todas las señales de audio. El principal inconveniente de este modelo reside en el hecho de que no está preparado para manejar de forma correcta los transitorios presentes en la señal. En esta sección se introduce el problema del modelado de transitorios, que complementa al modelado sinusoidal consiguiendo un modelo eficiente y adaptado a señales de audio genéricas de gran ancho de banda.

A continuación, se razona la necesidad de un modelo explícito para la parte transitoria de la señal, y se resumen las técnicas utilizadas en la literatura para el modelado de transitorios.

3.2.1. La necesidad de un modelado de transitorios

Como se ha visto anteriormente, el modelo sinusoidal extrae las componentes tonales que tienen una lenta variación en el tiempo, dejando como residuo aquellas componentes de señal que no cumplen con esta premisa, como son las componentes transitoria y ruidosa de la señal [George87][Serra89]. En cualquier caso, extrayendo un gran número de tonos mediante el modelado sinusoidal, es posible obtener las otras componentes, como hace la Transformada de Fourier, aunque no es la manera más eficiente, puesto que para representar un transitorio, corto en el tiempo, con tonos de larga duración se necesitan un gran número de frecuencias. En [George87] el modelado sinusoidal está pensado para aplicarlo sobre la señal de voz. En este caso, el residuo formado por transitorios y ruido es normalmente enmascarado por la componente tonal de la señal en los segmentos sonoros de ésta. Sin embargo, en aplicaciones de señal de audio de alta calidad este residuo es necesario para la integridad perceptual de la señal. Codificadores posteriores [Serra89], diseñados para señal de audio, incluyen un modelo explícito para el residuo. Este modelo es tan simple como un ruido blando filtrado en tiempo y frecuencia a partir de las características del residuo. Sin embargo, esta técnica no se ajusta a las características de la componente transitoria de la señal, lo que produce que los ataques se dispersen en el tiempo y se perciban como una señal ruidosa. Por lo tanto, debido a que los transitorios no se pueden modelar con alta calidad y de forma eficiente con los modelos sinusoidal y de ruido, es necesario la inclusión de un modelo de transitorios que maneje de forma independiente esta componente de la señal de audio.

3.2.2. Tipos de modelado de transitorios existentes

El primer método considerado para el tratamiento independiente de los transitorios se basa en separar las áreas transitorias del residuo. Para las partes no transitorias, se implementa un modelo de ruido, mientras que la componente transitoria se codifica por transformada, ya sea usando la transformada discreta de coseno [Levine98] o la transformada wavelet-packet [Ali95]. Aunque este método tiene buenos resultados perceptuales, se aleja de la codificación paramétrica

en tanto en cuanto no se tiene un modelo de transitorios sino su transformada codificada. Además, este enfoque no permite la modificación de la señal de audio en el dominio codificado.

La necesidad de un modelo paramétrico de baja complejidad para los transitorios, que permita un amplio rango de modificaciones de señal y se complemente con los modelos sinusoidal y de ruido existentes, ha motivado la aparición de diferentes enfoques para la solución del problema planteado. Sin embargo, muchos de estos métodos no consiguen sintetizar de forma satisfactoria los ataques de la mayoría de los instrumentos musicales.

Inversión de la envolvente del transitorio y modelado sinusoidal. Debido a que la componente transitoria de la señal se caracteriza por un breve intervalo de alta energía, una estrategia para parametrizarla consiste en la obtención de la envolvente [Brinker02] [MPEG03]. Tras esto, se divide la señal entre la envolvente, obteniéndose un segmento de características estacionarias en energía, el cual se puede modelar mediante la herramienta de extracción tonal para conseguir un residuo blanqueado. La envolvente se modela utilizando la función de Meixner de tiempo discreto [Brinker95]. La ventaja de esta función es que define dos parámetros: uno para describir la cresta del ataque y otro para modelizar su caída exponencial. Los resultados demuestran que la envolvente definida de esta forma se ajusta correctamente a diferentes ejemplos de transitorios que aparecen en las señales de audio [Brinker02]. Con este modelo de transitorios se generan los siguientes parámetros:

- La posición donde comienza el transitorio.
- Los dos parámetros de la envolvente definida como función de Meixner.
- Los parámetros sinusoidales (amplitud, frecuencia y fase) que describen la señal bajo la envolvente.

El principal problema de este método es que la suposición de que la señal bajo la envolvente se puede modelizar con un modelo tonal no es del todo cierta. Para obtener un modelo de transitorios siguiendo este enfoque, con resultados psicoacústicamente satisfactorios, son necesarios un elevado número de tonos en el modelo sinusoidal subyacente y, por lo tanto, un régimen binario elevado. Los resultados subjetivos conseguidos a regímenes binarios reducidos (24 kbits/s) mediante pruebas de audición [Brinker02] muestran que en las señales con transitorios, como las castañuelas, la calidad obtenida es baja, obteniéndose para estas señales mejores resultados con codificadores por transformada.

Aplicación de un modelo sinusoidal sobre el dominio transformado. La dualidad tiempo-frecuencia tiene propiedades que, utilizadas convenientemente, pueden ayudar al desarrollo de un modelo de transitorios basándose en el modelado sinusoidal ya conocido [Verma99]. La premisa a tener en cuenta es que el modelado sinusoidal es capaz de describir señales tonales, de lenta variación en el tiempo, debido a que corresponden a picos en el dominio de la frecuencia. Como se ha visto anteriormente, mediante la transformada de Fourier en cada segmento de señal (STFT) se realiza un análisis de seguimiento de picos espectrales, definiendo los tonos importantes de la señal de audio. Sin embargo, los transitorios también se definen como picos, si bien en el dominio del tiempo. Al contrario que las señales tonales, su transformada de Fourier da lugar a oscilaciones suaves del espectro. Por lo tanto, ha-

ciendo el mismo seguimiento de picos, en el dominio adecuado, se puede llegar a obtener un modelo de transitorios similar al modelado sinusoidal previamente estudiado.

En [Verma98] se utiliza la transformada discreta de coseno (DCT) como dominio sobre el que se aplica el modelo sinusoidal. La estructura del modelo de transitorios se basa en aplicar primero la transformada discreta de coseno (DCT) para, posteriormente, hacer un seguimiento de picos espectrales sobre la transformada de Fourier de segmentos consecutivos (STFT) de señal en el dominio de la DCT. De esta forma, se consigue, según [Verma98], una descripción precisa basada en parámetros de la parte transitoria de la señal. Sin embargo, los resultados obtenidos no han sido muy relevantes, obteniendo calidades aceptables a regímenes binarios del orden de 32 kbits/s, donde hay codificadores por transformada con calidad mayor, como, por ejemplo, el AAC [MPEG97a]. La razón de estos resultados hay que buscarla en la idoneidad del método propuesto para la extracción de transitorios. Así, en el modelo sinusoidal cada pico espectral se sustituye en el tiempo por un tono, suponiéndose que ese pico espectral se corresponde con el máximo de la transformada de la ventana utilizada desplazada a la frecuencia del tono modelado. Esta suposición es válida porque la señal de audio es muy tonal es los segmentos estacionarios. En este modelo de transitorios, esta suposición aplicada para segmentos transitorios en el dominio de la DCT no tiene por qué ser cierta. Muy al contrario, los picos que se obtienen aplicando la transformada de Fourier sobre la señal en el dominio DCT tienen una naturaleza muy diversa como corresponde a los diferentes tipos de transitorios. En cualquier caso, éste es el primer intento de modelado de transitorios que, tras estudiar las características propias de la parte transitoria del audio, aplica un modelo paramétrico que se basa en esos resultados.

Descomposiciones atómicas con átomos de Gabor o exponenciales amortiguadas. De entre los diferentes métodos propuestos para el modelado de transitorios, el empleo de sinusoides amortiguadas exponencialmente (Exponentially Damped Sinusoids, EDS) es la elección con mayor éxito encontrada en la literatura [Nieuwenhuijse98] [Goodwin97b], debido a la presencia de estas funciones en las señales de carácter oscilatorio. Una función de este tipo corresponde a la respuesta al impulso de un polo complejo, lo cual es una característica idónea para representar transitorios, especialmente aquellos producidos por sistemas lineales [Goodwin97]. Se han aportado diferentes soluciones para el desarrollo de un modelo de transitorios; sin embargo, el trabajo mejor documentado [Goodwin97] utiliza una descomposición atómica basada en matching pursuits [Mallat93] como método para extraer las características de la señal. Una mayor discusión acerca de estas descomposiciones se tratará en el siguiente capítulo. El algoritmo matching pursuits utiliza un diccionario sobre-completo compuesto por átomos, que no son más que funciones a extraer de la señal. En cada iteración del algoritmo se extrae el átomo más correlado (el que extrae más energía) con la señal. El diccionario se diseña para extraer ciertas características de la señal. Así, si se compone de exponenciales complejas [Verma99], se utiliza para modelado sinusoidal. En el caso de querer modelar transitorios, el diccionario se ajustará a este propósito cuando se componga de funciones exponenciales amortiguadas (EDS). En cada iteración del algoritmo se extrae una función de la familia EDS caracterizada por tres parámetros: factor de amortiguamiento, frecuencia y tiempo de comienzo.

Si bien la parametrización conseguida con este método es completa y los resultados prom-

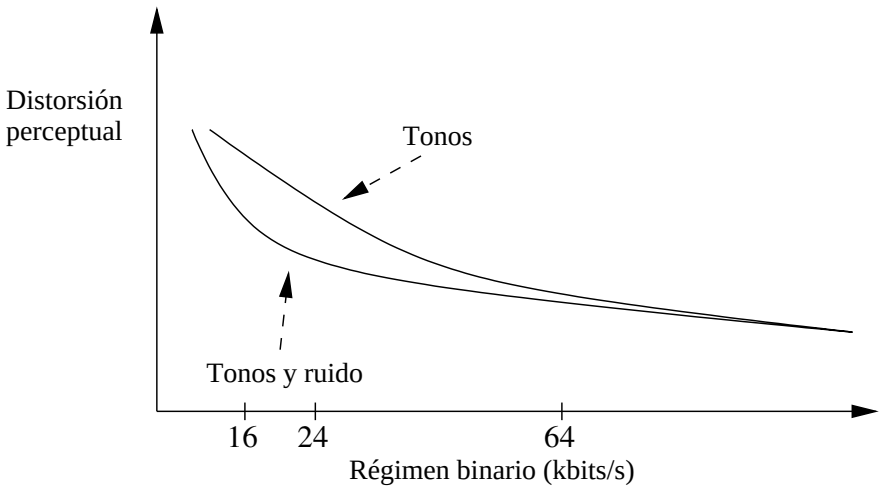

Figura 3.5: *Esbozo de la distorsión perceptual en relación al régimen binario cuando se utiliza sólo el modelado sinusoidal o el modelado sinusoidal más un modelo de ruido.*

etedores [Goodwin97], el principal problema radica en la gran complejidad necesaria para la obtención de los parámetros del modelo, debida, principalmente, al enorme tamaño del diccionario necesario. Aunque se ha trabajado sobre el tema aplicando bancos de filtros recursivos [Goodwin97b] para la actualización de las correlaciones del algoritmo matching pursuits [Mallat93], la complejidad sigue siendo demasiado elevada para obtener descomposiciones en tiempo real [Vera04a].

3.3. Modelado de ruido

Como se ha visto anteriormente, un segmento estacionario de la señal de audio se puede descomponer en un conjunto de tonos mediante la aplicación del modelado sinusoidal. Sin embargo, por muchos tonos que se incluyan en la descripción de la señal la calidad obtenida no es satisfactoria sin la inclusión de un modelo de ruido. La explicación de este hecho hay que buscarla en que tanto la componente sinusoidal como el ruido son dos entidades fundamentales en el campo de la percepción. Por ejemplo, el sonido producido por una flauta contiene estos dos elementos. Por un lado hay un conjunto de tonos relacionados armónicamente, y por otro, está el silbido (o ruido) producido por la corriente de aire. En el caso de la señal de voz, hay una distinción entre fonemas sonoros (tonales) y sordos (ruidosos). Si no se incluye el modelo de ruido, el resultado de la señal codificada no es natural, en cualquiera de los dos ejemplos. Esto conduce a la necesidad de incluir un modelo de ruido para codificar la señal residual derivada de la aplicación del modelo sinusoidal y del modelo de transitorios. La figura 3.5 ilustra que la descomposición de una señal de audio en un modelo tonal más un modelo ruidoso da como resultado una señal codificada de mayor calidad que cuando sólo se utiliza el modelo tonal, especialmente a bajo régimen binario.

Los modelos de ruido utilizados en la literatura se ocupan de parametrizar la envolvente de la señal en tiempo y frecuencia, ignorando la forma de onda como tal. Tanto la resolución espectral como la temporal del modelo deberían ajustarse a las propiedades de percepción de ruido del

oído humano. El modelo de ruido en general es común a las diversas propuestas encontradas [Goodwin96] [Fitz95] [Ding97] [Lam99] [Purnhagen99]. Así, en el decodificador, lo primero que se hace es generar ruido blanco, de donde se deriva el nombre de modelo estocástico asignado con frecuencia al modelo de ruido. En la literatura aparece un ejemplo donde incluso se genera una excitación multipulso [Ding97]. Esta señal estocástica se conforma después en tiempo y frecuencia. La conformación temporal se consigue típicamente dividiendo la señal de entrada en tramas cuyo tamaño se aproxima a la resolución temporal del oído humano [Goodwin97], si bien esta resolución depende de la frecuencia de la señal (4 ms a 1 kHz, 1 ms a 4 kHz) [Schijndel99]. Para la conformación espectral, se obtiene la envolvente espectral de la señal, aunque en este caso hay un gran número de opciones en la literatura. A continuación, se tratan más detenidamente las técnicas más utilizadas para modelar el ruido en frecuencia.

3.3.1. Esquemas de modelado de ruido basados en predicción lineal

El primer intento de parametrizar la envolvente del espectro del residuo corresponde a [Serra89], quien propuso una aproximación del espectro del residuo en segmentos lineales. Sin embargo, el mismo autor [Serra89] considera que el mejor método para obtener la envolvente del residuo se basa en la utilización de un filtro con respuesta al impulso infinita (Infinite impulse response, IIR) todo polos. Utilizando la señal residual como entrada, un sistema de predicción lineal (Linear Predictive Coding, LPC) optimiza los coeficientes del filtro [Markel76]. Una vez conseguidos estos coeficientes, el filtro necesario en el decodificador se debe implementar mediante una estructura en celdas (o *lattice*) para evitar problemas de estabilidad.

Otros autores han usado la técnica LPC para modelizar la componente residual [Serra97] [Purnhagen00]. Incluso, en algunos trabajos [Brinker00], se han utilizado filtros con ceros y polos (Auto-Regressive Moving Average, ARMA), pese al enorme crecimiento en complejidad del esquema de predicción lineal. Pese a todo, el mayor inconveniente de los modelos LPC es que la resolución lineal en frecuencia que obtienen no coincide con la resolución logarítmica (en escala de bark) del sistema auditivo humano [Zwicker90]. Una variante que solventa este problema es el uso de la técnica *warped*-LPC [Strube80], en la que se aumenta la resolución en bajas frecuencias a expensas de reducirla en altas frecuencias, de modo similar a como opera el oído humano. Este modelo LPC de frecuencia modificada debe tener una resolución similar a la escala de bark [Smith95] [Harma01], con el objetivo de seguir principios psicoacústicos. La técnica *warped*-LPC ha sido aplicada en codificación de audio en diversas ocasiones [Schuijers02] [Myburg04], debido a sus ventajas: 1) la resolución espectral de la predicción lineal se ajusta a la del oído humano, de tal forma que el error de cuantificación de los coeficientes tiene una distribución similar al umbral de enmascaramiento en frecuencia [Harma00a], 2) la complejidad necesaria para obtener los coeficientes del filtro lineal, así como su para su implementación, es baja. Una alternativa a la técnica *warped*-LPC es el uso de LPC con filtros de Laguerre [Brinker03], cuya estructura y resultados son similares. La validez de este método ha provocado su inclusión en los estándares de codificación paramétrica de audio de MPEG-4 [Schuijers02] [MPEG03].

3.3.2. Esquemas de modelado de ruido basados en filtros perceptuales

Otra vía de obtener la envolvente en frecuencia de la componente residual de audio consiste en diseñar un banco de filtros cuya respuesta en frecuencia tenga en cuenta las características del

oído humano. Este tipo de banco de filtros se diseña basándose en el hecho de que el oído humano es sensible al conjunto de la energía de ruido en cada banda crítica, independientemente de su distribución dentro de cada banda crítica [Zwicker90]. Teniendo esto en cuenta, si el residuo es ruido de banda ancha, se puede generar una réplica, indistinguible psicoacústicamente, a partir de ruido blanco filtrado, de forma que cada banda del filtro establezca la energía original de la señal residual en cada banda crítica. Sin embargo, el elevado número de bandas críticas hace que el régimen binario de un modelo de este tipo, que obtenga como parámetros la potencia de ruido de cada banda crítica, sea demasiado elevado. Para poder llevar a la práctica un modelo con estos principios, es necesario reducir el número de bandas del banco de filtros. Siguiendo este criterio, en [Goodwin97] se utiliza el concepto de ERB (Equivalent Rectangular Bandwidth) [Goodwin96], que proviene del concepto de bandas críticas, en el sentido de que el ancho de banda de cada banda es proporcional al de la banda anterior y crece en frecuencia. Este enfoque ha sido utilizado en algunos codificadores de audio [Verma99], porque es particularmente apropiado para la compresión de la señal.

3.4. Codificadores paramétricos

En este apartado se revisarán las características y herramientas utilizadas en los diversos codificadores paramétricos encontrados en la literatura. Como es lógico, el uso de herramientas paramétricas en los codificadores de audio ha sido un proceso evolutivo, es decir, se han ido incorporando modelos paramétricos para las componentes de la señal de audio a codificadores por transformada previos, con el objetivo de optimizar su funcionamiento a bajo régimen binario.

3.4.1. Codificadores híbridos

La primera aparición de herramientas paramétricas se produce con la utilización del modelo tonal en codificadores por transformada, dando lugar a codificadores híbridos. En estos codificadores, en la mayoría de los casos, se incluye el modelado sinusoidal como un mecanismo de análisis cuando la señal tiene propiedades estacionarias. En algunos casos, además, se ha usado la sustitución de ciertas bandas de la transformada con características estocásticas por ruido generado sintéticamente. Estos dos modelos paramétricos son los más usados por los codificadores híbridos. A continuación, se detalla el funcionamiento de los codificadores híbridos más relevantes.

Codificador de Ali [Ali95]

El primer codificador de audio que introduce tanto el modelado sinusoidal como la sustitución de bandas de señal con propiedades estocásticas por ruido es el descrito en la tesis de Ali [Ali95]. En la figura 3.6 se observa el esquema de funcionamiento de este codificador, donde el uso de transformadas se reduce, básicamente, al tratamiento de los transitorios por una transformada wavelet.

A grandes rasgos, el codificador lo primero que realiza es una análisis tonal que elimina de la señal esta componente produciendo un residuo. El modelo tonal está basado en el desarrollado por [Mcaulay86] para señal de voz, aunque no se utiliza el algoritmo de minimización por

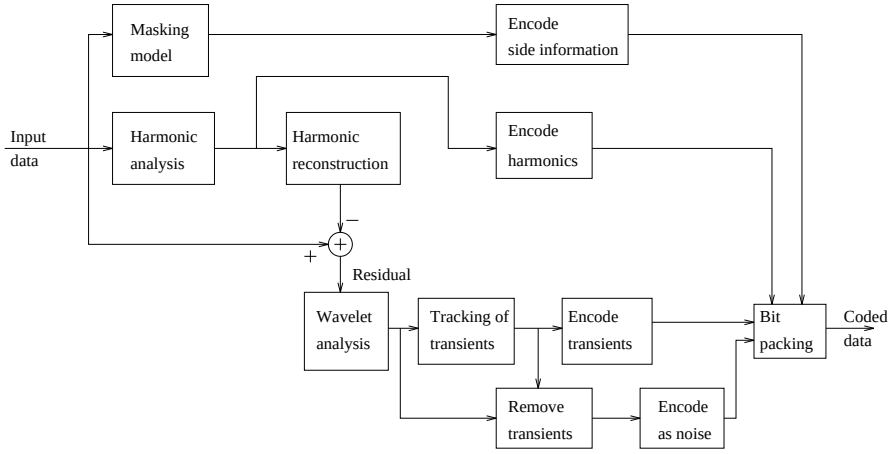

Figura 3.6: *Esquema del funcionamiento del codificador híbrido propuesto en [Ali95].*

mínimos cuadrados para la detección tonal. En este caso, se hace uso de una técnica, no utilizada previamente en audio sino en aplicaciones geofísicas, descrita en [Thomson82]. Al residuo obtenido, que contiene transitorios y ruido, se le aplica una transformada wavelet. Se consideran como ruidosas aquellas bandas wavelet por encima de 11 kHz, codificándose como ruido. Los parámetros codificados son la potencia de cada banda y la caída exponencial de su distribución de amplitudes. Las demás bandas se consideran transitorios y se cuantifican directamente los coeficientes de la transformada.

En cuanto a resultados, el codificador de Ali obtiene un régimen binario de 1 bit/muestra, que equivale a 44 kbits/s, suponiendo señales mono con calidad CD. El autor etiqueta su codificador de alta calidad o casi transparente, lo que equivaldría a un valor de 4 en la escala MOS, según la recomendación BS.1116-1 del ITU-R. Además, se admite que la señal con peor resultado subjetivo es la correspondiente a las castañuelas, que está incluida dentro del conjunto de señales de test para codificación de MPEG [MPEG03b], debido a la aparición de pre-ecos en la señal decodificada .

Codificador de Levine [Levine98]

Este codificador, propuesto por [Levine98] en su tesis doctoral, tiene como finalidad conseguir ciertos tipos de procesado de señal en el dominio comprimido, como son: modificaciones en el pitch y en la escala temporal. Para ello, es necesario conseguir separar las diferentes componentes (transitorios, tonos y ruido) de la señal de audio. Por ejemplo, para el cambio de tempo, hay que realizar la modificación de la duración de los tonos y el ruido de la señal en el tiempo, mientras que los transitorios sólo deben ser trasladados en el eje temporal. Como se observa en la figura 3.7, para conseguir esta separación, un determinado segmento de la señal de audio se etiqueta como transitorio o como estacionario. Si se tiene un segmento transitorio, se aplica codificación por transformada, mientras que si el segmento es estacionario, se aplica un modelo tonal seguido de otro de ruido. El codificador proporciona las herramientas necesarias para que un cambio de etiquetado no produzca distorsiones en la señal codificada.

En cuanto a las técnicas utilizadas, la codificación por transformada se realiza mediante la

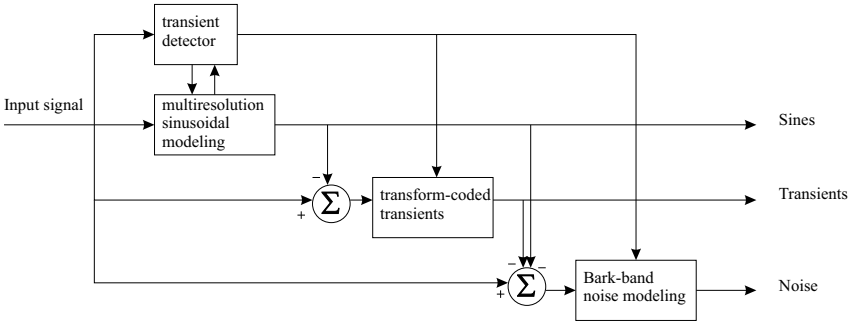

Figura 3.7: *Esquema del funcionamiento del codificador híbrido propuesto en [Levine98].*

transformada de coseno modificada MDCT, con un tamaño de ventana muy reducido, puesto que se aplica sólo en segmentos transitorios de la señal. Para realizar el modelado sinusoidal, se aplica un modelo multi-resolución, con tamaños de ventana decrecientes en la frecuencia para adaptarse a las características del oído. Una característica destacable de este codificador es que el modelado sinusoidal se aplica por debajo de 5 kHz, suponiendo que por encima de esta frecuencia la señal se puede modelar mediante ruido. La estimación de las frecuencias se realiza de la misma forma que en el codificador de Ali, mediante estimación de máxima semejanza [Thomson82]. En cuanto al modelo de ruido, se utiliza un modelado en banda de bark, es decir, se calcula en el codificador la potencia en cada banda de bark, sintetizándose el residuo en el decodificador mediante un banco de filtros con esta distribución en frecuencia.

En cuanto al análisis de resultados, el codificador de Levine está diseñado para proporcionar un régimen binario entre 20 y 32 kbits/s. En lo relativo a la calidad subjetiva, como afirma el autor, se introducen necesariamente algunos artefactos de codificación. En esta tesis se proporcionan una serie de ficheros codificados a 32 kbit/s para comparar los resultados entre AAC y este codificador. Con los ficheros elegidos no se aprecian excesivas diferencias, aunque la calidad del codificador AAC es algo mejor. En cualquier caso, es de esperar que al reducirse el régimen binario, el codificador de Levine tenga un menor descenso de calidad perceptual que el codificador AAC, al tratarse este último de un codificador por transformada.

3.4.2. Codificadores completamente paramétricos

Como se ha visto, los codificadores híbridos dejan entrever las posibilidades de la codificación paramétrica. Sin embargo, el paso a un codificador completamente paramétrico se puede considerar demasiado ambicioso, por el simple hecho de que en los comienzos de la codificación paramétrica no existía un modelo apropiado para manejar los transitorios de señal. Como consecuencia, en las señales donde está presente esta componente, los codificadores por transformada obtienen mejores resultados frente a los paramétricos. Además, hay que tener en cuenta que, en un codificador completamente paramétrico, los modelos de señal deben complementarse. Por ejemplo, el modelo sinusoidal no debe dejar componentes tonales en el residuo, aunque estos tonos no se escuchen, porque si se sintetizan como ruido el resultado perceptual es bastante pobre [Heusdens02b]. Como contrapartida, la completa parametrización de la señal de audio permite la realización, de manera directa y en el dominio comprimido, de modificaciones en la señal como

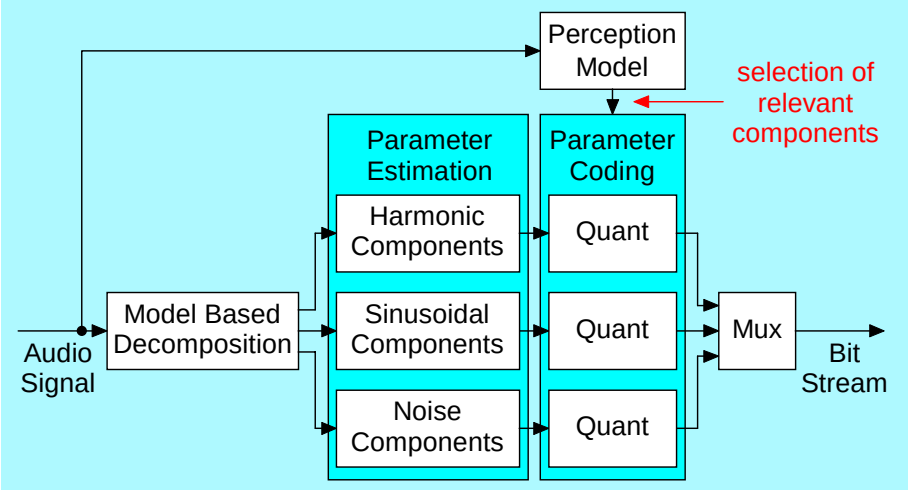

Figura 3.8: *Esquema del funcionamiento del codificador paramétrico HILN [Purnhagen00].*

el cambio de pitch o de tempo. Se realiza, a continuación, un estudio más pormenorizado de los codificadores paramétricos que han sido estandarizados por MPEG-4.

Codificador HILN

El codificador HILN [Purnhagen00] es el primer codificador de audio paramétrico estandarizado, ya que está incluido en la versión 2 de MPEG-4 [MPEG99c]. La señal de entrada se descompone en los siguientes parámetros, como se observa en la figura 3.8:

- Tonos sin relación armónica o tonos individuales. Se modelan con su amplitud y frecuencia.
- Tonos con relación armónica. Se describen por su frecuencia fundamental, amplitud y envolvente espectral de los parciales. Para todas las componentes tonales se utiliza el esquema de caminos tonales de [Mcaulay86] con una estimación por máxima semejanza [Purnhagen02].
- Ruido. Se modela por su amplitud en el tiempo y su envolvente en frecuencia. En este caso, la envolvente frecuencial es parametrizada mediante predicción lineal (LPC).

Adicionalmente, cuando un transitorio está presente, se incluyen los parámetros que describen la envolvente de la señal, siendo esta simple herramienta el modelo de transitorios usado. Debido a que el régimen binario para el que está diseñado va de 6 a 16 kbits/s, sólo se puede transmitir un pequeño conjunto de parámetros. Por lo tanto, se produce una selección de parámetros mediante criterios perceptuales. Respecto a la definición inicial, el codificador HILN ha sido mejorado [Purnhagen00], introduciendo cuantificación entrópica, cuantificación dependiente de criterios psicoacústicos e, incluso, escalabilidad en régimen binario.

En cuanto a los resultados perceptuales, aunque pobres, se pueden comparar a los proporcionados por los codificadores por transformada estandarizados a similar régimen binario. Esta afirmación se verifica analizando la figura 3.9, donde se incluyen los resultados de HILN no escalable a 6 kbits/s, HILN escalable a 6 kbits/s, AAC con TwinVQ (cuantificación vectorial) a 6 kbits/s, HILN no escalable a 16 kbits/s, HILN escalable a 16 kbits/s y AAC a 16 kbits/s. Notar

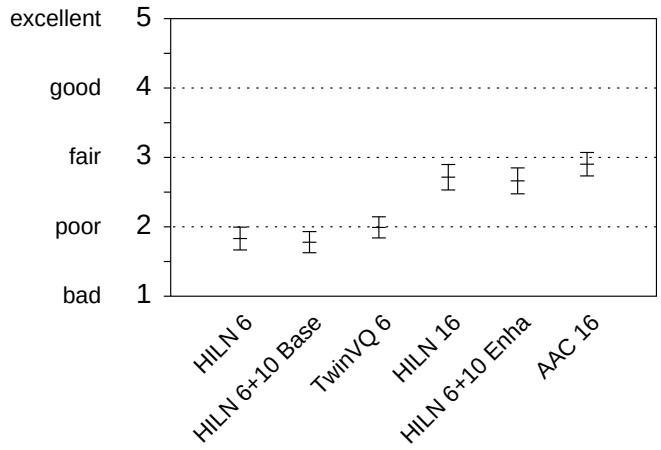

Figura 3.9: *Resultados de los test subjetivos para el codificador HILN. Para el mismo régimen binario, se ha usado HILN no escalable y escalable. Se comparan codificación HILN no escalable a 6 kbits/s (HILN 6), codificación HILN escalable a 6 kbits/s (HILN 6+10 Base), Twin VQ a 6 kbits/s, HILN no escalable a 16 kbits/s (HILN 16), codificación HILN escalable a 16 kbits/s (HILN 6+10 Enha) y AAC a 16 kbits/s. [Purnhagen00].*

que la señal de entrada tiene un ancho de banda de 8 kHz. Es destacable que la calidad desciende sólo un poco cuando se introduce la escalabilidad.

Codificador paramétrico de Philips (PPC)

El codificador PPC ha sido estandarizado en la versión 3 de MPEG-4. La necesidad de este codificador surge para aplicaciones de audio con alta calidad y un régimen binario muy reducido. Ante este problema, MPEG realizó una llamada a propuestas [MPEG01] para estandarizar un codificador paramétrico con mejor calidad que AAC a un régimen binario alrededor de 24 kbits/s. Pese a que el objetivo inicial era muy ambicioso, al final del proceso se ha conseguido estandarizar un codificador completamente paramétrico.

La tecnología utilizada por este codificador viene limitada por la complejidad, de forma que las técnicas elegidas obtienen una buena relación calidad/complejidad. Para describir el funcionamiento del codificador PPC, se detallan las técnicas utilizadas para cada una de las componentes del audio:

- Los transitorios se clasifican en dos tipos: transitorios de paso que no son tratados de forma especial, sólo se segmenta la señal en ese instante temporal; y transitorios propiamente dichos, los cuales son parametrizados mediante la envolvente de Meixner [Brinker95] y un modelado sinusoidal.

- Los tonos son extraídos, sin una búsqueda iterativa sino de forma paralela, siguiendo un algoritmo que minimiza una función de coste no perceptual similar a la optimización no lineal restringida [Hamdy99]. Se emplea un análisis multi-resolución, aunque no se agrupan las componentes armónicas [Myburg04].

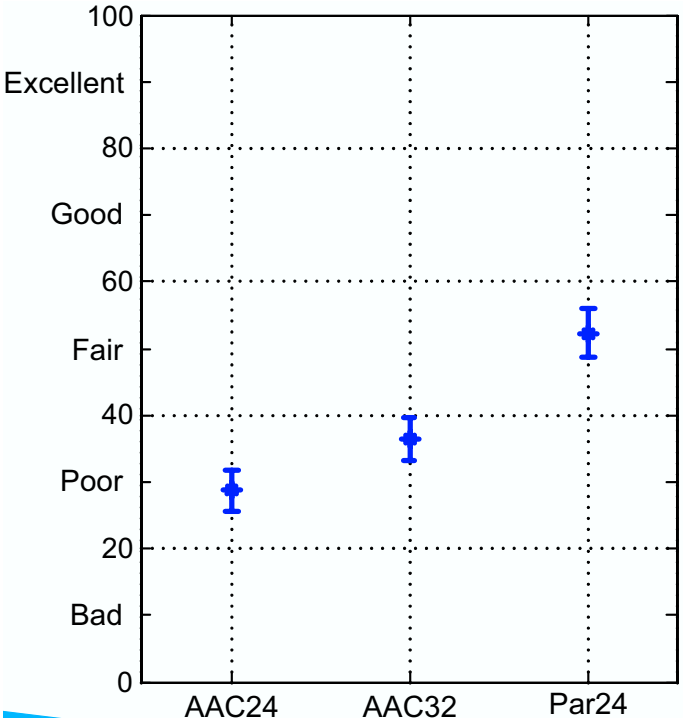

Figura 3.10: *Resultados de los test subjetivos para el codificador PPC. Se comparan los resultados a 24 kbit/s de PPC (Par en la figura) con AAC [Breebaart04].*

- El modelo de ruido se implementa mediante un filtro LPC con frecuencia modificada o filtro *warped-LPC*, empleando para ello filtros de Laguerre [Schuijers02].

Si bien el codificador se diseñó en un primer momento para codificación a 24 kbits/s en mono [Myburg04], posteriormente se incluyeron las señales estéreo. El método utilizado para tratar las señales estéreo emplea el codificador mono y una serie de parámetros para las relaciones entre canales, resultando un régimen binario adicional de 0,7 a 8 kbits/s [Breebaart04] dependiente de la calidad deseada. Finalmente, el codificador se optimizó para obtener 24 kbits/s en estéreo [Schuijers04].

Los resultados no han llegado a ser todo lo satisfactorios que se deseaba en un principio. Si bien, como aparece en la figura 3.10, la calidad según el test MUSHRA [ITU-R01] es superior en media, empleando las señales de prueba de MPEG, que la obtenida con AAC. Para algunas muestras de audio, como las señales vocales o las castañuelas, la calidad es sensiblemente menor [Brinker02].

3.4.3. Codificadores paramétricos escalables

Una de las mayores ventajas de la codificación paramétrica es la consecución, en base a información psicoacústica, de la escalabilidad en régimen binario. En este sentido, la codificación paramétrica, como se ha visto anteriormente, permite ordenar los tonos (e incluso las bandas de ruido) en una escala perceptual. De esta forma, si es necesario reducir el régimen binario, la codificación paramétrica proporciona buenos resultados perceptuales, limitando la distorsión introducida por el codificador a aquellas componentes con menor peso perceptual. Un codificador

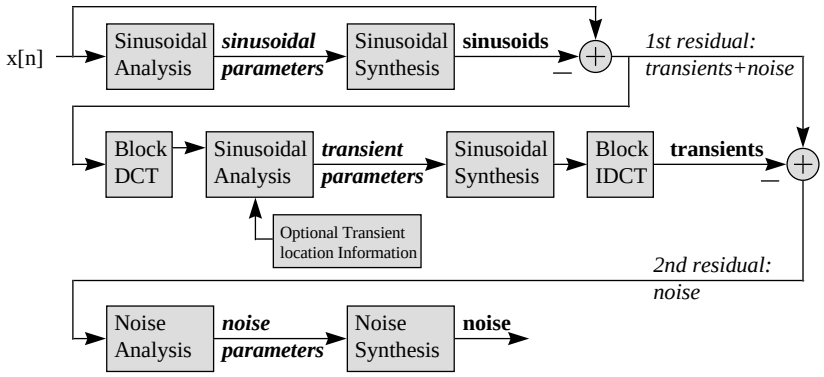

Figura 3.11: *Esquema del funcionamiento del codificador paramétrico de Verma [Verma99].*

escalable es deseable desde varios aspectos. La reducción del régimen binario, con una lenta y progresiva reducción de la calidad, puede permitir una robustez del codificador muy útil en la práctica. Por ejemplo, el codificador se puede adaptar de antemano a las características de la red o medio de transmisión utilizado, enviando la señal de audio con la mayor calidad posible. Un escenario para este funcionamiento es la telefonía móvil. Aún más, un codificador escalable puede ser utilizado de forma que se adapte en tiempo real a los requerimientos de la red utilizada, especialmente en internet. Para lograr este objetivo, el codificador usado debe cambiar el régimen binario de forma fina y sin incrementar la complejidad, de modo que se pueda usar en un escenario a tiempo real.

A continuación, se detallan las propuestas que aparecen en la bibliografía acerca de codificadores paramétricos escalables. Estos codificadores han sido diseñados especialmente para lograr alta escalabilidad en régimen binario, consiguiendo un ajuste fino del régimen binario, si es necesario.

Codificador de Verma [Verma99]

El codificador propuesto por Verma en su tesis doctoral [Verma99] es una adaptación del codificador de Levine para incluir escalabilidad en régimen binario. En este sentido, el autor realiza un esfuerzo para clasificar los diferentes parámetros de audio según su importancia perceptual, de forma que se consiga una pérdida lenta de la calidad conforme se reduce el régimen binario en el codificador. Además, Verma realizó un importante esfuerzo en la actualización de los algoritmos de extracción de parámetros de la señal, destacable sin duda al introducir el primer mecanismo de parametrización de los transitorios.

El esquema general del codificador de Verma se introduce en la figura 3.11. Como se observa, es un codificador paramétrico clásico de transitorios, tonos y ruido. Para la extracción de transitorios, Verma introduce la aplicación del modelo sinusoidal sobre el dominio de la transformada de coseno [Verma98], eliminando por completo la codificación por transformada que empleaba Levine. El modelo tonal se mejora mediante el empleo del algoritmo *matching pursuits* [Mallat93] con un diccionario de exponenciales complejas. En el caso del modelo de ruido, Verma reduce el número de bandas en frecuencia en el banco de filtros de Levine, hablándose ahora de un banco de filtros ERB.

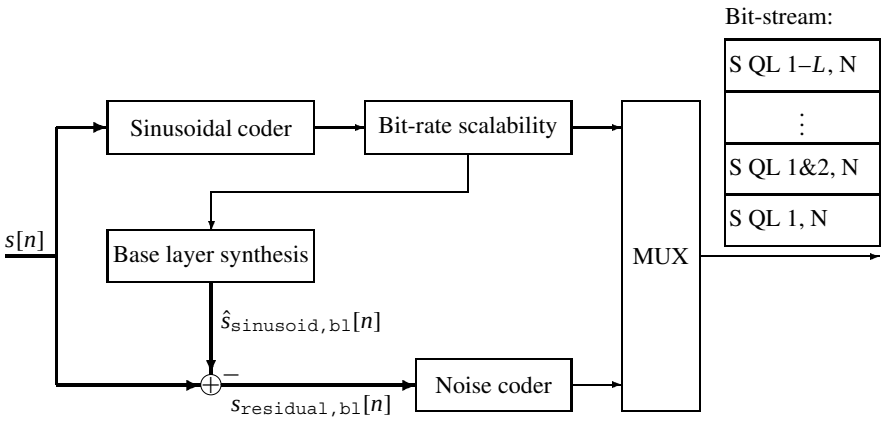

Figura 3.12: *Esquema del funcionamiento del codificador paramétrico de Myburg [Myburg04].*

El régimen binario obtenido varía desde 6 a 80 kbits/s, aunque sólo a partir de 16 kbits/s el fichero codificado contiene las tres componentes de la señal de audio. Según el autor, el codificador escalable propuesto a 80 kbits/s obtiene una calidad similar al codificador AAC de MPEG-4 a 64 kbits/s [Verma00], mientras que cuando el régimen binario se reduce a 16 kbits/s la calidad de ambos codificadores es muy similar. Como consecuencia, la calidad obtenida por el codificador de Verma es algo menor que la alcanzada por AAC para todos los regímenes binarios, excepto cuando se baja a 16 kbits/s, pero teniendo la ventaja de ser un codificador escalable.

Codificador de Myburg [Myburg04]

El codificador implementado por Myburg [Myburg04] es una adaptación del codificador PPC de Philips para conseguir escalabilidad. En este sentido, el autor estudia, además de la escalabilidad en régimen binario, la escalabilidad en complejidad del codificador para su uso en aplicaciones en tiempo real. El resultado es que, básicamente, el codificador de Myburg simplifica o elimina algunas de los algoritmos utilizados en el codificador PPC de Philips.

El esquema de funcionamiento del codificador de Myburg se esboza en la figura 3.12. La principal particularidad del codificador radica en que nunca se descarta como no audible ninguna parte (de la energía total) de la señal de audio. Esto se consigue enviando al codificador siempre los parámetros del ruido correspondientes al régimen binario menor o capa básica (*basic layer*). Los tonos enviados dependen del régimen binario objetivo y están separados por capas. Para conseguir el ruido final en el decodificador, se elimina de la potencia de ruido (de la capa básica) la potencia de los tonos de las capas superiores. Como consecuencia, la potencia de ruido sintetizado para capas superiores (alto régimen binario) es la resultante de restar a la potencia de la capa básica la potencia de los tonos de las capas superiores. De esta forma, nunca se elimina potencia de señal; eso sí, a bajo régimen binario, habrá información tonal generada por el modelo de ruido, que generará una distorsión ruidosa en el decodificador.

En relación a las técnicas utilizadas para implementar los modelos de señal, se producen algunas variaciones con respecto al codificador PPC de Philips. La extracción tonal es simplificada para reducir la complejidad, aplicando una extensión del método de Gauss-Newton [Myburg01]

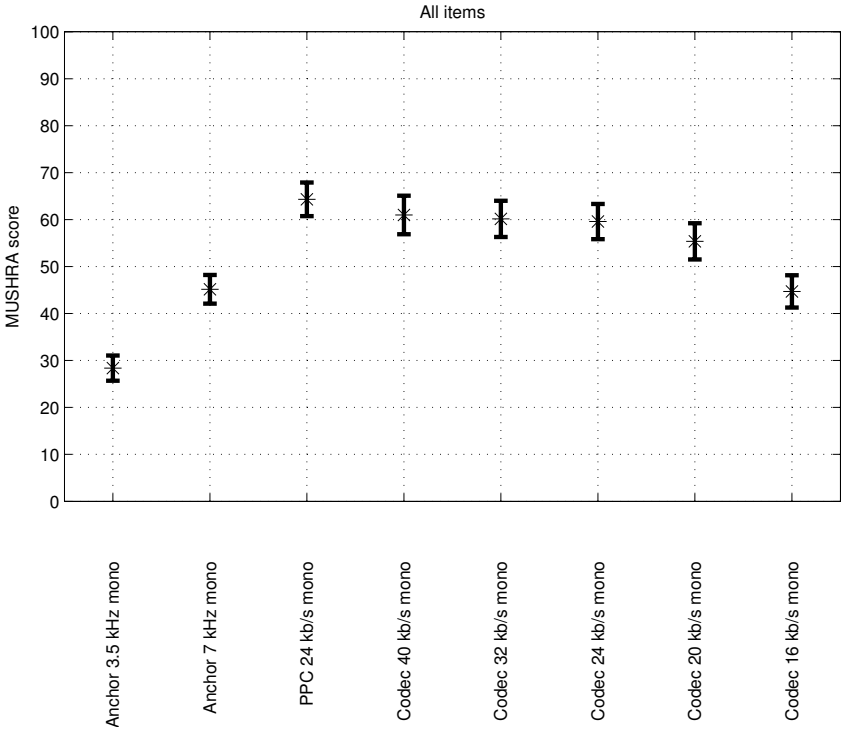

Figura 3.13: *Calidad perceptual obtenida por el codificador de Myburg a diferentes regímenes binarios. Las señales de comparación son una señal filtrada paso bajo a 3,5 kHz, otra filtrada a 7 kHz y la codificada mediante PPC. Se representa el valor medio y el intervalo del 95 % de confianza [Myburg04].*

para la extracción de forma paralela de los tonos. Además, se incluye un agrupamiento de los tonos en conjuntos armónicos con la finalidad de reducir el régimen binario. El modelo de ruido no cambia sustancialmente, pero el principal inconveniente del codificador es la eliminación del tratamiento para los transitorios. Si bien se realiza con la idea de reducir al máximo la complejidad, el impacto que tiene en la calidad perceptual hace cuestionable esta decisión.

Un resumen de la calidad obtenida por el codificador de Myburg se presenta en la figura 3.13. Como se puede observar, el principal inconveniente de este codificador es que, debido a la ausencia de un modelo de transitorios, la calidad perceptual no crece cuando se incrementa el régimen binario por encima de 24 kbits/s, estando siempre por debajo de la calidad del codificador PPC de Philips. Cuando el régimen binario se reduce de 24 a 16 kbits/s, también se reduce la calidad, pero de una forma progresiva. Este intervalo de funcionamiento es, por tanto, el idóneo para el codificador de Myburg.

Capítulo 4

Descomposiciones atómicas

4.1. Introducción

La representación de señales en función de átomos tiempo-frecuencia es un tema de interés desde su introducción por Gabor en los años 40 [Gabor46]. La noción fundamental de los modelos atómicos es que una señal se puede descomponer en funciones elementales localizadas en tiempo-frecuencia. Estas descomposiciones son muy útiles para aplicaciones como análisis y codificación.

La descomposición de una señal en funciones es un problema complejo que tiene un gran número de soluciones. Si una señal $x[n]$ se descompone en un conjunto de funciones $g_m[n]$, el modelo de señal queda de la forma,

$$x[n] = \sum_{i=1}^{I} \alpha_{m(i)} \cdot g_{m(i)}[n] \tag{4.1}$$

donde $\alpha_{m(i)}$ son los coeficientes asociados a cada función $g_{m(i)}[n]$ y el número de funciones es I. De forma general, para tratamiento digital de la señal, la longitud de la señal $x[n]$ ha de ser finita, de valor N. Para la mayoría de las aplicaciones de tratamiento de señal se utilizan descomposiciones basadas en transformadas, como las de Fourier o wavelet. En estos casos, las funciones forman una base y el número de funciones para descomponer la señal es $I = N$. Una forma de expresar el método de cálculo de estas transformadas es mediante la notación matricial,

$$\mathbf{x} = \mathbf{D}\boldsymbol{\alpha} \tag{4.2}$$

donde $\mathbf{x}$ es un vector columna $(N \times 1)$, que representa la señal, $\boldsymbol{\alpha}$ un vector columna $(N \times 1)$ de coeficientes y $\mathbf{D} = [\mathbf{g}_1\, \mathbf{g}_2 \ldots \mathbf{g}_m \ldots \mathbf{g}_N]$ una matriz $(N \times N)$, cuyas columnas son los vectores columna $\mathbf{g}_m$ de funciones. El cálculo de los coeficientes, cuando $\mathbf{D}$ es invertible, viene definido por,

$$\boldsymbol{\alpha} = \mathbf{D}^{-1}\mathbf{x} \tag{4.3}$$

Cuando las funciones en $\mathbf{D}$ forman una base biortogonal se cumple $\mathbf{D}^{-1} = \tilde{\mathbf{D}}^H$, donde el superíndice H denota conjugada y traspuesta, y $\tilde{D}$ es la base dual de D. Cuando la base es ortogonal la base dual es simplemente $\tilde{D} = D$. En este caso, los coeficientes individuales se calculan como,

$$\boldsymbol{\alpha}_m = \mathbf{g}_m^H \mathbf{x} = < \mathbf{x}, \mathbf{g}_m > \tag{4.4}$$

Debido a esta fácil manera de calcular la descomposición, las transformadas han sido ampliamente usadas en tratamiento de la señal. El principal problema de las transformadas es que no proporcionan un modelo que aproxime una señal arbitraria en unos pocos coeficientes transformados [Goodwin97]. Para aplicaciones de análisis y codificación es importante modelizar la señal a partir de un número muy reducido de funciones, que representen las características principales de la señal a tratar. En este sentido, se ha trabajado con transformadas para determinar aquella base que mejor represente la señal en este sentido. Este algoritmo conocido como la selección de la mejor base (Best Orthogonal Basis, BOB) [Coifman92] ha sido utilizado en los últimos años en la codificación de audio con bases wavelet packet. En codificación de audio, se ha utilizado la medida de la entropía perceptual para seleccionar la mejor base [Ruiz01]. Sin embargo, para la señal de audio, que es una señal excepcionalmente variable, se consigue mediante descomposiciones atómicas una solución más compacta que la que se obtiene mediante la selección de la mejor base.

Para resolver los inconvenientes de las transformadas, mejorando la capacidad de compresión de la descomposición, una solución es incrementar el número de funciones en que se puede descomponer la señal. Aplicando este principio, cuando el número de funciones del diccionario $\mathbf{D}$ es $M > N$, queda una matriz $(N \times M)$ en la ecuación 4.2, y no se puede hablar ya de base, sino que se dice que el diccionario es sobrecompleto . El diseño del diccionario debe incorporar, a priori, un amplio rango de comportamientos tiempo-frecuencia, para adaptadarse a las características de la señal a tratar. El principal problema de las descomposiciones sobrecompletas es el método de cálculo de los coeficientes [Davis94]. En este caso, ya no es deseable el uso de la matriz inversa del diccionario $\mathbf{D}^{-1}$, puesto que la matriz $\mathbf{D}$ no es singular, lo que quiere decir que la solución no es única. Como consecuencia, han aparecido varios métodos para el cálculo apropiado de una descomposición atómica. En cualquier caso, la selección de aquellas funciones o átomos y el cálculo asociado de sus coeficientes es un problema no lineal, donde se busca que unos pocos átomos describan a grandes rasgos el comportamiento de la señal. El principal inconveniente de las descomposiciones atómicas es la alta complejidad asociada a su cálculo, lo que limita en gran medida su uso.

Un problema derivado de la complejidad en el cálculo de las descomposiciones atómicas es la necesaria limitación del tamaño del diccionario. Así, conociendo la señal a tratar, o realizando un preprocesamiento de la misma, se utiliza un tipo de diccionario u otro, con el fin de extraer las características de la señal en un tiempo de computación razonable. En el campo del audio, esto se traduce, por ejemplo, en etiquetar cada segmento de la señal de audio con un preprocesado sencillo que clasifique la señal en transitoria o tonal. En base a este valor, se puede elegir un diccionario formado por señales impulsivas u otro formado por señales sinusoidales. Más adelante, se tratará con detenimiento el diseño del diccionario.

A continuación, se revisan los métodos más usuales para la obtención de las descomposiciones atómicas, con sus ventajas e inconvenientes, así como algunos ejemplos de su funcionamiento.

4.2. Métodos de cálculo

En general, las diferentes estrategias para seleccionar las funciones y calcular los coeficientes de éstas, con el fin de obtener una descomposición atómica, se pueden clasificar en dos categorías principales:

- Métodos iterativos. Como primera aproximación, el cálculo de una descomposición atómica se puede realizar siguiendo un proceso iterativo, de tal forma que, en cada iteración, se elija el átomo y su coeficiente asociado o peso. La selección iterativa de los átomos se debe regir por un proceso de optimización de una medida determinada. Se pueden encontrar varios ejemplos en la literatura que siguen este esquema, aunque todos son derivaciones del algoritmo iterativo conocido como *matching pursuits* (MP) [Mallat93]. Es importante tener en cuenta que, aunque se elija el átomo óptimo en cada iteración, un algoritmo de este tipo conduce a una solución subóptima, ya que no se considera una optimización conjunta de todos los átomos elegidos. Sin embargo, el uso de estos algoritmos se ha extendido en muchas aplicaciones de tratamiento de señal, debido a que permiten el cálculo de una descomposición atómica con una complejidad razonable y unos resultados satisfactorios [Goodwin97].
- Métodos paralelos. En este caso, no se restringe la búsqueda de los átomos a seleccionar un átomo en cada iteración del método, sino que se realiza de forma paralela, es decir, se optimizan el conjunto de los átomos seleccionados de foma simultánea, conduciendo, por tanto, a soluciones óptimas. Para cada método, se define la medida sobre la señal que se desea optimizar, aunque la filosofía sea la misma para todos ellos. Pese a encontrar soluciones óptimas globalmente, este tipo de métodos han sido escasamente utilizados, debido principalmente a que la ventaja que aportan no compensa el incremento de complejidad necesario para su implementación.

Seguidamente, se detalla el procedimiento que siguen los métodos más destacados que se encuentran en la bibliografía.

4.2.1. Métodos paralelos

Los métodos paralelos se caracterizan generalmente por llegar a soluciones óptimas basadas en una medida global a optimizar y por su gran carga computacional. El nombre de métodos paralelos es debido a que en el cálculo de la solución se determina a la vez el valor de todos los coeficientes.

Método de tramas (MOF)

El método de tramas (Method Of Frames, MOF) [Daubechies88] escoge, de entre todas las soluciones, aquella en la que los coeficientes en su conjunto tienen norma l^2 mínima, es decir, en la que los coeficientes tienen energía mínima. La notación matricial del problema queda expresada como,

$$min||\boldsymbol{\alpha}||^2 \quad para \quad \mathbf{D}\boldsymbol{\alpha} = \mathbf{x} \tag{4.5}$$

La solución a este problema $\boldsymbol{\alpha}^{MOF}$ es única y es la solución de longitud mínima. Por lo tanto, desde un punto de vista geométrico, de todas las soluciones posibles, que pertenecen al espacio multidimensional que representan los átomos del diccionario, la solución adoptada por el método MOF es aquella que está más cerca del origen (en distancia euclídea).

Una ventaja del método MOF es que existe una manera directa de calcular la descomposición [Daubechies88], mediante la matriz pseudo-inversa de Moore Penrose $\mathbf{D}^+$, que vale $\mathbf{D}^+ = \mathbf{D}^H(\mathbf{D}\mathbf{D}^H)^{-1}$. Con esta matriz, se calcula la solución de mínima longitud de un sistema de ecuaciones lineales de la forma,

$$\boldsymbol{\alpha}^{MOF} = \mathbf{D}^+\mathbf{x} = \mathbf{D}^H(\mathbf{D}\mathbf{D}^H)^{-1}\mathbf{x} \tag{4.6}$$

Debido a esta forma sencilla de cálculo, el método de tramas es computacionalmente poco complejo, aunque los resultados que obtiene no son adecuados. La descomposición obtenida por el método de tramas tiene dos inconvenientes principales [Chen96]:

- El número de átomos con un coeficiente distinto de cero es elevado, es decir, no se reduce la solución a unos pocos átomos. Esto es debido a que la medida utilizada no penaliza a los átomos con un coeficiente de valor bajo, puesto que estos valores no incrementan significativamente la energía. Como resultado, en la solución suelen aparecer todos los átomos con una correlación con la señal distinta de cero.

- La resolución tiempo-frecuencia es limitada. La causa de este problema hay que buscarla en el método de cálculo de la solución, estando la resolución limitada por el operador $\mathbf{D}\mathbf{D}^H$. Esto hace que, aunque la señal a descomponer esté formada sólo y exclusivamente por uno de los átomos del diccionario, la pérdida de resolución al aplicar el operador mencionado, provoca que la energía se expanda entre muchos átomos correlados con el buscado.

Como demostración de las dos afirmaciones anteriores, se incluyen las figuras 4.1 y 4.2. En primer lugar, en la figura 4.1 se dibuja un plano tiempo-frecuencia o plano de fase ideal [Daubechies88] de una función wavelet-packets. A continuación, tomando como señal a analizar esta función formada por un átomo del diccionario, que en este caso es un conjunto de funciones wavelet packets, se dibuja en la figura 4.2 el plano tiempo-frecuencia de la solución ideal y la obtenida por el método de tramas o MOF. Como se observa, la solución adoptada por el método MOF no se reduce a un sólo átomo, como sería deseable, sino que la energía se ha dispersado entre muchos de los átomos correlados con el que se ha construido la señal.

Basis pursuits

La definición del método Basis pursuits (BP) [Chen95] está basada en la del método MOF, en el sentido de que la solución se basa también en la minimización de una norma de los coeficientes, salvo que en este caso la norma utilizada es el valor absoluto de la amplitud o norma l^1, quedando matricialmente,

$$min||\boldsymbol{\alpha}|| \quad para \quad \mathbf{D}\boldsymbol{\alpha} = \mathbf{x} \tag{4.7}$$

Este método resuelve el principal problema del MOF, al cambiar la norma. Ahora, los coeficientes con valores pequeños tienen una penalización mayor y tienden a desaparecer de la

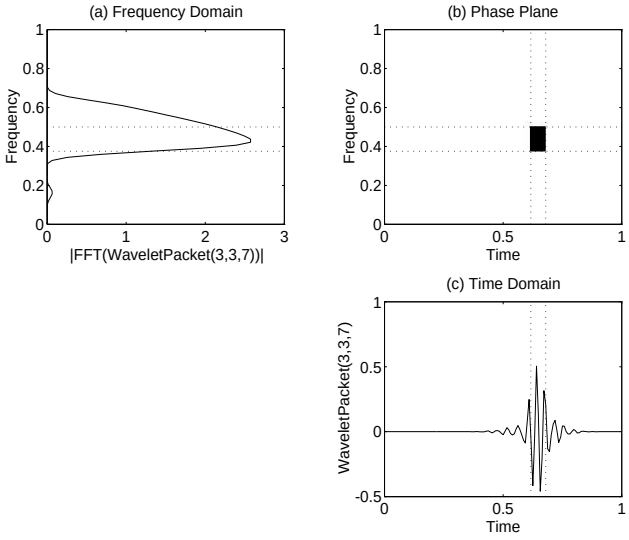

Figura 4.1: *Plano de fase ideal de una función wavelet-packets. Figura obtenida mediante el toolbox* atomizer *de Matlab disponible en la dirección de Internet* http://www-stat.stanford.edu/~atomizer/*.*

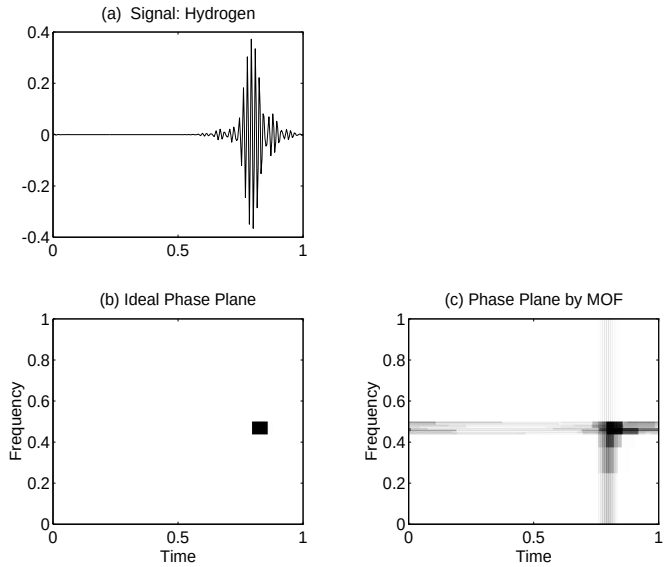

Figura 4.2: *Ejemplo de funcionamiento del método de tramas o MOF. Figura obtenida mediante el toolbox* atomizer *de Matlab.*

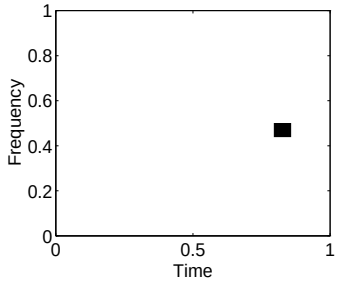

Figura 4.3: *Ejemplo de funcionamiento del método basis pursuits (BS) para una señal formada por un átomo wavelet packets. Se dibuja el plano de fase de la solución mediante BS con un diccionario wavelet packets que incluye al átomo de la señal. Figura obtenida mediante el toolbox* atomizer *de Matlab.*

solución, quedando como solución, para el caso de la señal formada por un sólo átomo, el mismo átomo. Una prueba de este hecho se puede observar en la figura 4.3, pudiendo afirmarse que este método tiene gran capacidad de concentración de la solución en unos pocos átomos [Chen96]. El principal inconveniente del BS radica en que los algoritmos a utilizar para calcular la descomposición son excesivamente complejos, sobre todo para aplicaciones de compresión.

La solución al problema de optimización planteado por el método BS es única, aunque no existe una formulación matemática asociada para su cálculo, es decir, se trata de un método heurístico. La descomposición mediante BS habría que calcularla, en el peor de los escenarios, siguiendo una búsqueda exahustiva. Esta no es una aproximación válida en la práctica, por lo que Chen al presentar BS en [Chen95] sugirió algunos de los algoritmos que pueden ayudar a encontrar la solución. En este sentido, en [Chen96] se revisan las relaciones del método BS con las estrategias de programación lineal, con el objetivo de encontrar un algoritmo cuya complejidad crezca linealmente (y no exponencialmente) con el tamaño de la señal de entrada. Como resultado, Chen propone usar dos algoritmos alternativos, cuyas diferencias se estudian a continuación:

- *Simplex* . Este algoritmo llega a la solución óptima a costa de una gran complejidad. El algoritmo comienza con una solución no óptima al problema de la minimización con norma l^1; por ejemplo, la solución del método MOF. A continuación, comienza un proceso iterativo en el que se cambia a otra solución (o conjunto de coeficientes) que reduce la norma. Existen reglas que garantizan la convergencia hacia la solución óptima [Gill91] y evitan procesos cíclicos. El proceso continua hasta que no hay mejora posible, es decir, cuando se halla la solución óptima.

- *Interior-point*. Este algoritmo es menos complejo y, además, se puede parar en cualquier momento, obteniéndose una solución subóptima reducida a un subgrupo de átomos. Básicamente, es una modificación del algoritmo anterior para reducir su complejidad. Ahora, se comienza considerando un sólo átomo en el diccionario; por ejemplo, el más correlado con la señal. Esto no es una solución al problema, porque el diccionario es menor que el tamaño de la señal. En cada iteración, se añade un átomo y se encuentran los coeficientes que minimizan la norma l^1 para ese número de átomos, por ejemplo vía simplex. La solución óptima se encuentra cuando se incluyen todos los átomos. Las ventajas de este método se hacen evidentes cuando el diccionario es capaz de modelizar con unos pocos átomos a la señal, ya que en este caso el tiempo de computación es reducido. Además, este método es

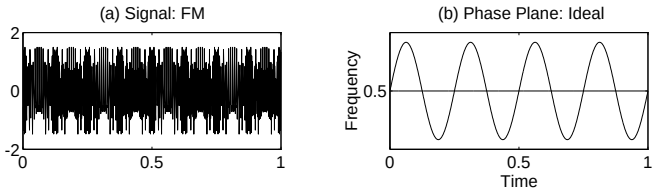

Figura 4.4: *Señal FM y su plano de fase ideal. Figura obtenida mediante el toolbox* atomizer *de Matlab.*

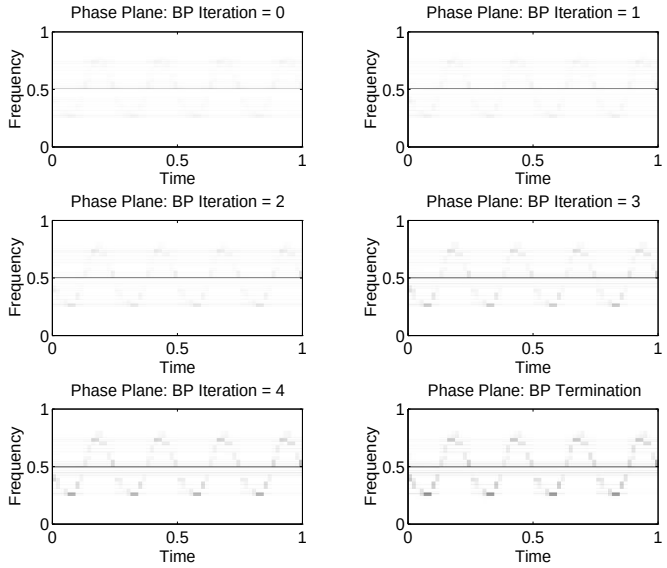

Figura 4.5: *Ejemplo de funcionamiento del algoritmo interior-point para el método basis pursuits (BS) para una señal FM con un diccionario de* cosine packets. *Figura obtenida mediante el toolbox* atomizer *de Matlab.*

el indicado cuando está limitado el tiempo de computación. Aunque se detenga el proceso de búsqueda antes de llegar a la solución óptima, se puede representar la mayor parte de la señal con unos pocos átomos. Está especialmente indicado en aplicaciones de *denoising*, determinando una parada anterior a la solución final, o en la búsqueda de los átomos que mejor representan la señal, por ejemplo, para la solución de direcciones de llegada [Chen96]. En la figura 4.4 se muestra una señal FM y su plano de fase ideal. Aplicando un diccionario de cosine packets y el algoritmo *interior-point*, el resultado de cada iteración se muestra en la figura 4.5. Se puede observar como el algoritmo es, por decirlo de alguna manera, embebido, porque en cada iteración va refinando el plano de fase, obteniéndose una idea del mismo desde las primeras iteraciones.

La implementación de ambos algoritmos para desarrollar el método BP, así como la posibilidad de elegir entre varios tipos de diccionario, se encuentra en una *toolbox* de Matlab en la dirección de internet: *http://www-stat.stanford.edu/~atomizer/*. Además, se incorpora en estos programas la posibilidad de obtener otro tipo de métodos para calcular descomposiciones atómicas, como el MOF y algunos más que se tratarán posteriormente.

La complejidad de los diferentes métodos es un factor importante a la hora de evaluar la

aplicación práctica de los mismos. En el caso del método MOF, la complejidad es de orden $O(Mlog(M))$ (donde M es el tamaño del diccionario) y viene determinada por la expresión 4.6. Sin embargo, la complejidad del método BS depende del algoritmo utilizado en el cálculo, de la señal de entrada y del tipo de diccionario. Así, lo más que se puede hacer en este caso es calcular el límite superior de complejidad del método. Aplicando el algoritmo *interior-point*, la complejidad máxima está limitada a $O(Mlog(M))$ por etapa. Teniendo en cuenta que el número de etapas puede llegar a ser tan grande como el tamaño del diccionario, la complejidad se torna prohibitiva. Como conclusión, cabe decir que el método *basis pursuits* sólo ha sido empleado con éxito en aplicaciones donde el número de átomos necesarios para modelizar la señal es limitado, con el objetivo de no incrementar demasiado la complejidad. Un ejemplo de este tipo de aplicaciones es la eliminación de ruido o *denoising* de la señal [Chen95].

FOCUSS

Como se ha dejado entrever en el método anterior, en muchas aplicaciones no es necesario encontrar una solución que describa la señal de entrada por completo, a partir de los átomos del diccionario, sino que basta con que los coeficientes encontrados aproximen, de forma suficiente, la señal de entrada

$$x[n] \approx \sum_{i=1}^{I} \alpha_{m(i)} \cdot g_{m(i)}[n] \tag{4.8}$$

o en notación matricial,

$$\mathbf{x} \approx \mathbf{D}\boldsymbol{\alpha} \tag{4.9}$$

Este sería el resultado, por ejemplo, de parar el método *basis pursuits* calculado mediante el algoritmo *interior-point* en una iteración intermedia. Además, desde un punto de vista práctico, debido a que las señales suelen estar contaminadas por ruido, no tiene mucho sentido encontrar soluciones exactas, porque se estará representando en la descomposición la parte ruidosa, y eso se pretende evitar en aplicaciones de análisis de señal.

Una vez tenido en cuenta que las soluciones prácticas pueden ser aproximaciones de la señal, se puede introducir la definición del método FOCUSS (FOCal Underdetermined System Solver), como aparece en [Gorodnitsky97]. Este método se utiliza cuando se supone que el número de átomos del diccionario para representar la mayor parte de la señal es pequeño con respecto al tamaño del diccionario. El método FOCUSS necesita una inicialización de los coeficientes de los átomos $\boldsymbol{\alpha}_0$, como ocurría en la implementación del algoritmo *simplex* para *basis pursuits*. Sin embargo, la inicialización de este método debe ser algo particular. Se deben dejar a cero los coeficientes de aquellos átomos de los que se tenga la constancia que no forman parte de la solución y evaluar sólo aquellos que puedan formar parte de la misma. Como se verá porteriormente, el algoritmo funciona con cualquier inicialización, pero si esta no es correcta, la complejidad crece demasiado. En cada iteración del algoritmo, el método FOCUSS implementa una minimización del subespacio que forman los I átomos con coeficientes iniciales distintos de cero. Sin embargo, a diferencia del método MOF, esta minimización está pesada por una matriz $\mathbf{W}_i$, por lo que se minimiza en cada iteración,

$$min||\mathbf{W}_i{}^{-1}\boldsymbol{\alpha}||^2 \quad para \quad \mathbf{D}\boldsymbol{\alpha} = \mathbf{x} \tag{4.10}$$

La solución a la expresión 4.10 se realiza de la misma forma que para el método MOF mediante la pseudo-inversa de Moore Penrose,

$$\boldsymbol{\alpha}_{i+1} = \mathbf{W}_i(\mathbf{D}\mathbf{W}_i)^+\mathbf{x} \tag{4.11}$$

En función de la definición que se haga de la matriz $\mathbf{W}_i$, y partiendo de $\boldsymbol{\alpha}_i$, se cambia el peso de los átomos en la minimización del error. Aunque puede ser otra cualquiera, la definición más extendida es [Lee87],

$$\mathbf{W}_i = diag(\boldsymbol{\alpha}_i) \tag{4.12}$$

por lo que, desde un punto de vista de señales, se minimiza en cada iteración la función,

$$\sum_{l=1,\,\alpha_i[l]\neq 0}^{M} \left(\frac{\alpha[l]}{\alpha_i[l]}\right)^2 \tag{4.13}$$

El método FOCUSS converge en el sentido de que lleva hacia cero aquellos átomos que no corresponden a la solución final y refuerza el subconjunto, dentro de los I inicializados, que describen la solución final. Se puede decir que el método FOCUSS focaliza la posible solución dentro de un subconjunto de átomos. Además, este método está exclusivamente diseñado para aplicaciones que concentren la solución sólo en un pequeño grupo de átomos. Una pregunta que surge en este momento es cuando parar el algoritmo. Las propuestas encontradas en la literatura [Gorodnitsky97] se limitan a detener el algoritmo cuando haya dos grupos claros de átomos, un grupo cercano a cero y otro con valores representativos, que formará la solución final.

Está claro que el método FOCUSS conduce a soluciones subóptimas, pero, en aplicaciones prácticas, estas son admisibles para reducir la complejidad. La medida de la carga computacional del algoritmo es sencilla, puesto que se puede deducir a partir del método MOF. Así, el orden de complejidad por iteración es $O(Ilog(I))$, siendo I el número de átomos distintos de cero en la inicialización $\boldsymbol{\alpha}_0$. Notar que la complejidad depende en gran medida de este valor inicial. Además, según las referencias [Lee87] [Gorodnitsky97], el número de iteraciones necesario para lograr un resultado satisfactorio es altamente dependiente de la elección de $\boldsymbol{\alpha}_0$. Se puede afirmar, visto su funcionamiento, que el método FOCUSS extiende un puente entre los métodos paralelos globales y los métodos iterativos, puesto que proporciona una solución subóptima, aunque en cada iteración optimice los átomos inicializados en su conjunto.

Las aplicaciones del método FOCUSS son variadas, pero se encuentra casi siempre relacionado en la bibliografía con la resolución de direcciones de llegada [Lee87] y del tratamiento de la señal eléctrica o magnética que produce el cerebro, señales EEG o MEG, respectivamante.

4.2.2. Métodos iterativos

Los métodos iterativos se caracterizan por obtener soluciones subóptimas en descomposiciones atómicas de señales y calcularse mediante métodos que permiten una complejidad reducida. La principal característica de los métodos iterativos es que deciden la elección de uno o varios átomos

(y sus coeficientes asociados) en cada iteración, manteniendo estos valores fijos para las siguientes iteraciones. Esta premisa permite reducir la complejidad, porque en el cálculo sólo se optimiza el átomo (o los átomos) a elegir en la iteración actual. Una consecuencia derivada de este tipo de cálculo es que las soluciones obtenidas sólo pueden aproximarse a la señal de entrada según la ecuación 4.8.

Matching pursuits

El primer método iterativo para la obtención de descomposiciones atómicas encontrado en la bibliografía es *matching pursuits* [Mallat93]. Posteriormente, han surgido una serie de modificaciones sobre la base de este método para intentar solucionar los inconvenientes que éste introduce. En cualquier caso, estos intentos no han tenido el éxito esperado, porque incrementan demasiado la carga computacional del método, que es una de sus grandes ventajas.

La definición del método *matching pursuits* es muy sencilla. Es un método iterativo que en cada iteración extrae de la señal el átomo que minimiza la energía del resto. Por lo tanto, en cada iteración i se extrae un átomo $\mathbf{g}_{m(i)}$ con su coeficiente (o peso) asociado $\alpha_{m(i)}$,

$$\mathbf{r}^i = \begin{cases} \mathbf{x} & i = 0 \\ \mathbf{r}^{i-1} - \alpha_{m(i)}\mathbf{g}_{m(i)} & i > 0 \end{cases} \tag{4.14}$$

Como se puede observar, el método inicializa la señal residuo, $\mathbf{r}^0$, a la señal de entrada $\mathbf{x}$, y trabaja para el resto de iteraciones con el residuo $\mathbf{r}^i$. Tanto la elección del átomo óptimo, como su coeficiente asociado, se determinan a partir de la minimización de la energía del residuo en cada iteración. En notación matricial quedaría,

$$\min_{\mathbf{g}_m \in \mathbf{D}} ||\mathbf{r}^i||^2 \quad para \quad \mathbf{r}^i = \mathbf{r}^{i-1} - \boldsymbol{\alpha}_m^i \mathbf{g}_m \tag{4.15}$$

donde los valores $\boldsymbol{\alpha}_m^i$ son los coeficientes asociados a cada uno de los elementos $\mathbf{g}_m$ del diccionario $\mathbf{D}$. De entre todos los valores $\boldsymbol{\alpha}_m^i$, se elige el que minimiza la norma l^2 de $\mathbf{r}^i$, que es el coeficiente del átomo óptimo, $\alpha_{m(i)}$. A partir de esta definición, se obtiene, además, la expresión para calcular los coeficientes $\boldsymbol{\alpha}_m^i$. El problema planteado se resuelve mediante la introducción del valor de $\mathbf{r}^i$ en el cálculo del mínimo.

$$\begin{array}{ll} \min_{\mathbf{g}_m \in \mathbf{D}} ||\mathbf{r}^i||^2 & = \\ \min_{\mathbf{g}_m \in \mathbf{D}} ||\mathbf{r}^{i-1} - \boldsymbol{\alpha}_m^i \mathbf{g}_m||^2 & = \\ \min_{\mathbf{g}_m \in \mathbf{D}} ||\mathbf{r}^{i-1}||^2 + |\boldsymbol{\alpha}_m^i|^2 ||\mathbf{g}_m||^2 - < \boldsymbol{\alpha}_m^i \mathbf{g}_m, \mathbf{r}^{i-1} > - < \mathbf{r}^{i-1}, \boldsymbol{\alpha}_m^i \mathbf{g}_m > & \end{array} \tag{4.16}$$

Como el vector $\mathbf{r}^{i-1}$ es una constante en la iteración i, en lugar de minimizar la función anterior, se puede maximizar,

$$\begin{array}{ll} \max_{\mathbf{g}_m \in \mathbf{D}} < \boldsymbol{\alpha}_m^i \mathbf{g}_m, \mathbf{r}^{i-1} > + < \mathbf{r}^{i-1}, \boldsymbol{\alpha}_m^i \mathbf{g}_m > - |\boldsymbol{\alpha}_m^i|^2 ||\mathbf{g}_m||^2 & = \\ \max_{\mathbf{g}_m \in \mathbf{D}} 2Re\{< \mathbf{r}^{i-1}, \boldsymbol{\alpha}_m^i \mathbf{g}_m >\} - |\boldsymbol{\alpha}_m^i|^2 ||\mathbf{g}_m||^2 & \end{array} \tag{4.17}$$

El valor de $\boldsymbol{\alpha}_m^i$ se halla simplemente obteniendo el valor máximo de la función anterior. La solución final se puede escribir como,

$$\max_{\mathbf{g}_m \in \mathbf{D}} ||\boldsymbol{\alpha}_m^i||^2 \quad para \quad \boldsymbol{\alpha}_m^i = \frac{< \mathbf{r}^{i-1}, \mathbf{g}_m >}{||\mathbf{g}_m||^2} \tag{4.18}$$

A partir de ahora, y sin pérdida de generalidad, se supondrá que los elementos del diccionario tienen energía unidad, es decir, $||\mathbf{g}_m||^2 = 1$. La ecuación inicial del método se puede re-escribir como,

$$\mathbf{r}^i = \begin{cases} \mathbf{x} & i = 0 \\ \mathbf{r}^{i-1} - < \mathbf{r}^{i-1}, \mathbf{g}_{m(i)} > \mathbf{g}_{m(i)} & i > 0 \end{cases} \tag{4.19}$$

escogiéndose en la iteración i el elemento del diccionario $\mathbf{g}_{m(i)}$ que maximice la función,

$$\mathbf{g}_{m(i)} = \arg \max_{\mathbf{g}_m \in \mathbf{D}} || < \mathbf{r}^{i-1}, \mathbf{g}_m > ||^2 \tag{4.20}$$

En vista de la ecuación 4.20, se comprueba que el método *matching pursuits* (MP) trabaja con las correlaciones como valor para obtener la descomposición. En cada iteración se escoge como átomo óptimo, y por lo tanto se elige en la descomposición, el átomo más correlado con el residuo. Se puede comprobar, observando la ecuación 4.4, que este método utiliza exactamente la misma medida que las transformadas, lo que redunda en el uso de algoritmos de cálculo rápidos, ya desarrollados para éstas. Aunque el principal problema parezca el cálculo de la correlación (que debe realizarse en cada iteración), se puede simplificar este cálculo relacionando las correlaciones entre iteraciones sucesivas. Así, aplicando la relación obtenida en la ecuación 4.19, se puede escribir,

$$< \mathbf{r}^i, \mathbf{g}_m > = < \mathbf{r}^{i-1}, \mathbf{g}_m > - < \mathbf{r}^{i-1}, \mathbf{g}_{m(i)} >< \mathbf{g}_{m(i)}, \mathbf{g}_m > \tag{4.21}$$

La actualización de correlaciones mediante este procedimiento limita el cálculo directo a la primera iteración, $< \mathbf{x}, \mathbf{g}_m >$. Para el resto de iteraciones, se utiliza la ecuación 4.21, que necesita tener almacenadas en memoria las correlaciones cruzadas entre todos los elementos del diccionario. A partir de estas correlaciones cruzadas y del peso asociado al átomo óptimo, $\alpha_{m(i)} = < \mathbf{r}^{i-1}, \mathbf{g}_{m(i)} >$, se implementa fácilmente el procedimiento de actualización de las correlaciones.

Otro aspecto significativo a tener en cuenta, al ser el coeficiente del átomo óptimo, $\alpha_{m(i)} = < \mathbf{r}^{i-1}, \mathbf{g}_{m(i)} >$, es que el residuo en la iteración i está incorrelado con el átomo óptimo $\mathbf{g}_{m(i)}$, es decir, se cumple el principio de ortogonalidad,

$$< \mathbf{r}^i, \mathbf{g}_{m(i)} > = 0 \tag{4.22}$$

Como consecuencia de esta propiedad, se puede decir que el método MP extrae la proyección del átomo óptimo en el espacio, como se observa en la figura 4.6.

Un resultado derivado a tener en cuenta es que la energía del residuo en la iteración i se puede expresar como,

$$||\mathbf{r}^i||^2 = ||\mathbf{r}^{i-1}||^2 + |\alpha_{m(i)}|^2 \tag{4.23}$$

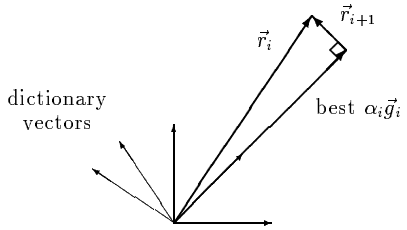

Figura 4.6: *Método* matching pursuits *y el principio de ortogonalidad [Goodwin97].*

resultado lógico al cumplirse el principio de ortogonalidad. Como consecuencia se puede demostrar [Mallat93] que el método converge, es decir, que la energía del residuo tiende asintóticamente a cero.

Para simplificar se resumirán los pasos del algoritmo,

- Inicialización:
  1. Se inicializa $\mathbf{r}^0 = \mathbf{x}$.
  2. Se calcula el valor inicial de las correlaciones $\boldsymbol{\alpha}_m^1 =< \mathbf{x}, \mathbf{g}_m >$
- Para cada iteración:
  1. Se escoge como función óptima aquella que da lugar al valor máximo de las correlaciones:
     $\mathbf{g}_{m(i)} = \arg \max_{\mathbf{g}_m \in \mathbf{D}} ||\boldsymbol{\alpha}_m^i||^2$ con su coeficiente asociado $\alpha_{m(i)}$
  2. Se actualizan las correlaciones $\boldsymbol{\alpha}_m^i = \boldsymbol{\alpha}_m^{i-1} - \alpha_{m(i)} < \mathbf{g}_{m(i)}, \mathbf{g}_m >$

La complejidad asociada al método *matching pursuits* siguiendo el algoritmo arriba descrito es la siguiente:

1. Para la inicialización, hay que calcular las correlaciones entre la señal y todos los átomos del diccionario. Este cálculo queda matricialmente $\boldsymbol{\alpha}^1 = \mathbf{D}^H \mathbf{x}$. De forma general, esta complejidad es de orden $O(Mlog(M))$ [Gribonval01]. Sin embargo, para la mayoría de los diccionarios, es posible encontrar algoritmos de cálculo eficiente de las correlaciones, como ocurre en el caso de un diccionario compuesto de exponenciales complejas o wavelet packets.

2. Para cada iteración, es necesario actualizar las correlaciones, lo cual tiene un orden de complejidad de orden $O(M)$.

El principal problema del algoritmo propuesto radica en la cantidad de memoria necesaria para guardar las correlaciones cruzadas entre todos los elementos del diccionario. En algunos casos, cuando el tamaño de esta memoria se hace demasiado elevado, es posible intercambiar memoria por complejidad, dejando la puerta abierta al cálculo directo de las correlaciones en todas las iteraciones, o mejor, a algoritmos mixtos de cálculo [Goodwin97]. En estos algoritmos mixtos, se actualizan las correlaciones a partir de algunas de las correlaciones cruzadas y algunos cálculos, gracias las propiedades específicas del diccionario elegido.

Como se observa analizando el coste computacional del algoritmo MP, éste crece con el número de átomos extraídos de la señal, por lo que es recomendable limitar el número de iteraciones. Por otro lado, cuantas más iteraciones se realicen se obtendrá una representación más exacta de la señal. Está claro que cuando el número de iteraciones sea alto, se puede escribir,

$$x[n] \approx \sum_{i=1}^{I} \alpha_{m(i)} \cdot g_{m(i)}[n] \tag{4.24}$$

El número de iteraciones I para detener el algoritmo depende de la señal a analizar, del diccionario y de la aplicación en cuestión. En general, es una suposición aceptable que la señal de entrada estará distorsionada por ruido. Así, cuando el número de iteraciones sea elevado y se halla extraído la mayor parte de la energía de la señal, el residuo no estará correlado con los elementos del diccionario y los coeficientes tendrán valores reducidos. Este hecho permite detener el algoritmo cuando se halla modelizado la mayor parte de la energía de la señal, o bien cuando los coeficientes a extraer estén por debajo de un determinado umbral.

Como ejemplo de funcionamiento se modela una señal formada por dos tonos próximos en frecuencia. En la figura 4.7 se observa la descomposición obtenida mediante las frecuencias elegidas con un diccionario tonal. Se puede observar como el primer valor es erróneo, puesto que no extrae ninguno de los dos tonos de la señal (en líneas punteadas), sino que extrae un tono intermedio. Esto es debido a que los dos tonos que forman esta señal tienen una correlación cruzada alta, siendo el tono más correlado con la señal un tono intermedio. El efecto que provoca esta situación es que el método MP se equivoca en una iteración temprana y después no se puede conseguir una descomposición adecuada.

Si bien la mayor ventaja es su reducido coste computacional, en el caso de una descomposición con pocos coeficientes el método MP tiene algunos inconvenientes, como son:

- En señales formadas por átomos del diccionario correlados entre sí, un error del método en iteraciones tempranas puede hacer que no se extraiga el átomo adecuado y se necesiten varias iteraciones adicionales para eliminar la energía de la señal. Este efecto se debe a que la medida utilizada es la mayor correlación con todos los átomos (y es un algoritmo iterativo).

- Aún tratando con señales formadas por átomos del diccionario no correlados, en el escenario habitual de empleo con un diccionario altamente sobre-completo, cuyos elementos representan densamente el espacio de la señal, se producen problemas cuando se implementan muchas iteraciones. En las primeras iteraciones, los primeros átomos elegidos en la descomposición tenderán a ser ortogonales entre sí. El resultado, desde un punto de vista geométrico, es que las sucesivas proyecciones de cada átomo elegido serán independientes. Sin embargo, en iteraciones posteriores, esta tendencia se invierte, llegándose a extraer átomos correlados con átomos elegidos anteriormente. Este problema, conocido como *re-admisión*, puede ponerse de manifiesto por una mala elección del umbral de parada del método.

Para solventar los problemas anteriores, han surgido en la bibliografía numerosas modificaciones sobre el método MP básico. Sin embargo, casi ninguno de ellos ha tenido un efecto práctico

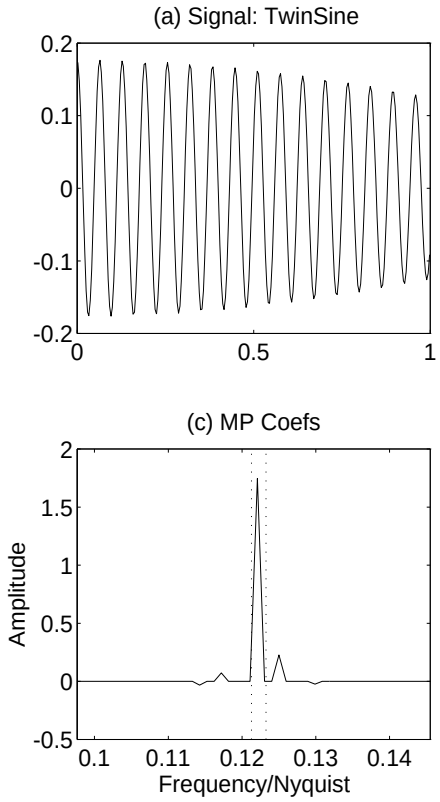

Figura 4.7: *Descomposición en un plano de fase de dos tonos próximos en frecuencia con el método MP. Figura obtenida mediante el toolbox* atomizer *de Matlab.*

importante, porque solucionan los problemas de MP a base de incrementar su complejidad. Se introducen brevemente estas modificaciones, en función de la causa que originó su propuesta.

1. Para evitar la re-admisión: este problema está generado por la correlación entre átomos elegidos en diversas iteraciones del diccionario. La solución más apropiada parece ser la ortogonalización de los elementos no elegidos del diccionario respecto al elemento extraído en cada iteración mediante el proceso de ortogonalización de Gram-Schmidt. Este método, conocido como *Orthogonal Matching Pursuits* (OMP) [Pati93] debido al proceso de ortogonalización, llega a una solución exacta en N iteraciones, es decir, en tantas iteraciones como longitud tenga la señal. Como contrapartida, OMP varía el contenido del diccionario, ya que las funciones del diccionario se ven modificadas en el proceso de cálculo de los coeficientes. Esto genera un inconveniente en algunas aplicaciones de análisis, porque al modificar los átomos pueden variar algunas de sus características. Otra desventaja es el incremento en la carga computacional asociado al proceso de ortogonalización. En la bibliografía han aparecido gran número de propuestas que intentan disminuir la carga computacional del método OMP [Chen95b] [Natarajan95] [Adler96] [Rebollo02].

2. Para evitar errores con átomos correlados: en este caso está claro que si se quiere evitar este problema lo más lógico sería utilizar métodos paralelos que no fijan un átomo por iteración, sino que re-calculan los mismos. Sin embargo, en la literatura aparece un método, conocido como *High Resolution Pursuits* (HRP) [Gribonval96][Jaggi98] que, diseñado en base a MP, intenta solucionar este problema. Un tratamiento más exhaustivo de este método se realizará posteriormente.

3. Para cambiar la medida de energía: la medida que utiliza *matching pursuits* para elegir el átomo que extrae en cada iteración es la energía. En algunas aplicaciones, este guiado por energía puede no ser el método más indicado. En este sentido, en aplicaciones de audio quizás sea más interesante extraer el átomo más importante perceptualmente. Se dice entonces que hay un guiado perceptual. Este campo ha sido explotado empleando el método MP con un diccionario de exponenciales complejas para mejorar el modelo sinusoidal. Así, se define una medida perceptual [Verma99b] [Heusdens02] que elige el tono más importante psicoacústicamente en cada iteración del algoritmo.

High resolution pursuits

Como se ha detallado anteriormente, el algoritmo *Matching Pursuits* (MP) optimiza la energía que se extrae en cada iteración. Esto, como se observa en la figura 4.7, produce errores cuando la señal de análisis incluye átomos correlados entre sí. La causa hay que buscarla en que MP realiza la elección del átomo óptimo optimizando la energía global de la señal, pero que no se adapta a las características locales de la misma. Este problema se puede solventar aplicando el algoritmo *Basis Pursuits*, pero la complejidad asociada a este algoritmo paralelo lo hacen prohibitivo.

Sin embargo, en [Jaggi98], se propone un algoritmo iterativo que, basado en MP y en correlaciones, permite, gracias a modificaciones en las mismas, adaptarse a la estructura local de la señal. Este algoritmo se conoce como *High Resolution Pursuits* (HRP) y exhibe una resolución cercana a la presentada por BP. Además, el método HRP tiene una complejidad similar al método MP, debido a que está basado en las mismas medidas. La diferencia radica en que sólo se tiene

en cuenta la correlación con los átomos del diccionario que tienen una alta resolución temporal (o frecuencial). Para el resto de átomos, con baja resolución, no se tiene en cuenta la correlación, sino la correlación cruzada con los anteriores, llamada similaridad . El algoritmo esta diseñado de forma que no se incrementa en ningún caso la energía local de la señal.

Para implementar el método HRP, se divide el diccionario en dos sub-diccionarios disjuntos $\mathbf{D} = \mathbf{D}_a \cup \mathbf{D}_b$, donde $\mathbf{D}_a$ representa los átomos con alta resolución temporal y $\mathbf{D}_b$ el conjunto de átomos con baja resolución temporal. Para los átomos del sub-diccionario con alta resolución, la medida para elegir el átomo óptimo es la correlación, como sucede en MP:

$$S(\mathbf{r}^{i-1}, \mathbf{g}_m) = \langle \mathbf{r}^{i-1}, \mathbf{g}_m \rangle, \quad \mathbf{g}_m \in \mathbf{D}_a \tag{4.25}$$

Para cada átomo del diccionario con baja resolución, no se puede utilizar directamente la correlación, porque puede modificar el comportamiento local de la señal. En su lugar, como se muestra a continuación, hay que utilizar la similaridad, que se va a definir mediante la correlación con los átomos de alta resolución y la correlación cruzada entre éstos y el átomo de baja resolución en cuestión. Para realizar este proceso, se define un subconjunto de átomos $\mathbf{g}_{\lambda_m} \in \mathbf{L}_m$ asociado a cada átomo con baja resolución $\mathbf{g}_m \in \mathbf{D}_b$. Los átomos del subconjunto $\mathbf{L}_m$ se eligen de entre los átomos de alta resolución que tienen un soporte temporal incluido en el soporte del átomo $\mathbf{g}_m$ y están modulados en la misma frecuencia.

En general, para los átomos pertenecientes al subconjunto $\mathbf{L}_m$, la correlación $\langle \mathbf{r}^i, \mathbf{g}_{\lambda_m} \rangle$ representa la cantidad de "energía" de $\mathbf{r}^i$ localizada en la sección tiempo-frecuencia de $\mathbf{g}_{\lambda_m}$. Por lo tanto, la medida que se utilice en HRP para representar el peso de $\mathbf{g}_m \in \mathbf{D}_b$ no debe dañar la energía local de la señal, es decir debe cumplir,

$$|\langle S(\mathbf{r}^{i-1}, \mathbf{g}_m)\mathbf{g}_m, \mathbf{g}_{\lambda_m} \rangle| \leq |\langle \mathbf{r}^{i-1}, \mathbf{g}_{\lambda_m} \rangle| \tag{4.26}$$

A partir de la ecuación (4.26), se deriva la nueva medida de similaridad $S(\mathbf{r}^{i-1}, \mathbf{g}_m)$ para el sub-diccionario de baja resolución del algoritmo HRP. Esta medida maximiza la cantidad de energía que se puede extraer al elegir el átomo $\mathbf{g}_m$ en la iteración i-ésima del algoritmo sin dañar la energía local:

$$S(\mathbf{r}^{i-1}, \mathbf{g}_m) = \varepsilon \cdot \min_{\mathbf{g}_{\lambda_m} \in \mathbf{L}_m} \frac{|\langle \mathbf{r}^{i-1}, \mathbf{g}_{\lambda_m} \rangle|}{|\langle \mathbf{g}_m, \mathbf{g}_{\lambda_m} \rangle|}, \quad \mathbf{g}_m \in \mathbf{D}_b \tag{4.27}$$

donde ε se incluye para no dañar a ninguno de los componentes del subconjunto $\mathbf{L}_m$. Aunque se escoja el valor mínimo de todos los valores del subconjunto, la elección es válida siempre que todas las correlaciones cruzadas tengan el mismo signo, dado ε se evalúa según:

- Si $\langle \mathbf{r}^{i-1}, \mathbf{g}_{\lambda_m} \rangle$ tiene el mismo signo para todos los átomos $\mathbf{g}_{\lambda_m} \in L_m$, entonces ε es el signo común.
- En otro caso $\varepsilon = 0$.

En MP, el producto interno usado no tiene en cuenta si el átomo elegido tiene o no una resolución adecuada, por lo que al utilizar esta medida se puede variar el comportamiento local de la señal. Sin embargo, en HRP se utiliza el producto interno sólo para los átomos con alta resolución, usando para los átomos con baja resolución la similaridad definida en la ecuación

(4.27). Esto evita crear energía donde no la había en la señal original y, además, permite distinguir características con alta resolución temporal.

La implementación del método se puede resumir, a partir de la ecuación inicial como,

$$\mathbf{r}^i = \begin{cases} \mathbf{x} & i = 0 \\ \mathbf{r}^{i-1} - S(\mathbf{r}^{i-1}, \mathbf{g}_{m(i)})\mathbf{g}_{m(i)} & i > 0 \end{cases} \tag{4.28}$$

Para elegir el átomo óptimo $\mathbf{g}_{m(i)}$ en la iteración i-ésima, es necesario calcular el valor de la medida de la similaridad para todos los átomos del diccionario $S(\mathbf{r}^{i-1}, \mathbf{g}_m)$. Este valor se calcula dependiendo de la resolución propia de cada átomo:

$$S(\mathbf{r}^{i-1}, \mathbf{g}_m) = \begin{cases} \langle \mathbf{r}^{i-1}, \mathbf{g}_m \rangle, & \mathbf{g}_m \in \mathbf{D}_a \\ \varepsilon \cdot \min_{\mathbf{g}_{\lambda_m} \in \mathbf{L}_m} \frac{|\langle \mathbf{r}^{i-1}, \mathbf{g}_{\lambda_m} \rangle|}{|\langle \mathbf{g}_m, \mathbf{g}_{\lambda_m} \rangle|}, & \mathbf{g}_m \in \mathbf{D}_b \end{cases} \tag{4.29}$$

escogiéndose en la iteración i el elemento del diccionario $\mathbf{g}_{m(i)}$ que maximice la función,

$$\mathbf{g}_{m(i)} = \arg \max_{\mathbf{g}_m \in \mathbf{D}} ||S(\mathbf{r}^{i-1}, \mathbf{g}_m)||^2 \tag{4.30}$$

El principal inconveniente asociado a HRP es la cuidadosa organización que ha de realizarse de los elementos del diccionario:

- Por un lado, es necesario separar los átomos del diccionario entre átomos con alta resolución y átomos con baja resolución para formar los conjuntos $\mathbf{D}_a$ y $\mathbf{D}_b$, respectivamente. Esta distinción habrá que realizarla en función de la resolución deseada y las características a determinar. De forma general, basta con determinar la resolución deseada y clasificar los átomos en base a esa elección. Sin embargo, no siempre es posible esta separación. Así, en el caso de la figura 4.7, no se puede obtener una resolución mayor en frecuencia, puesto que los átomos son todos tonos con la misma resolución.

- Por otro lado, la selección del conjunto de átomos de alta resolución $\mathbf{g}_{\lambda_m} \in \mathbf{L}_m \subset \mathbf{D}_a$ relacionados con cada átomo de baja resolución $\mathbf{g}_m \in \mathbf{D}_b$ es una cuestión de diseño muy comprometida, puesto que los resultados del algoritmo pueden variar mucho en función de esta elección. En [Vera03a], se demuestra que no es necesario incluir en el subconjunto $\mathbf{g}_{\lambda_m} \in \mathbf{L}_m$ todos los átomos de alta resolución con correlación cruzada no nula con el átomo $\mathbf{g}_m \in \mathbf{D}_b$, sino que incluyendo sólo los máximos locales de la correlación cruzada, se puede reducir, como se verá posteriormente, la complejidad del algoritmo. Sin embargo, este procedimiento es altamente dependiente de la naturaleza de las correlaciones cruzadas entre los elementos del diccionario.

Un ejemplo de funcionamiento donde el HRP demuestra sus ventajas debe incluir una señal formada por átomos altamente correlados. Así, en la figura 4.8 se analiza la descomposición obtenida por BP, MP y HRP de una señal formada por cuatro átomos: una delta en el tiempo, un tono de alta frecuencia y dos átomos wavelet packets altamente correlados entre sí. En esta figura, el diccionario utilizado es un diccionario formado por funciones wavelet-packet, tonos y deltas. Como se observa en la figura, el algoritmo HRP obtiene resultados similares que el BP, pudiendo descomponer la señal en los cuatro átomos que la forman, mientras que en MP los

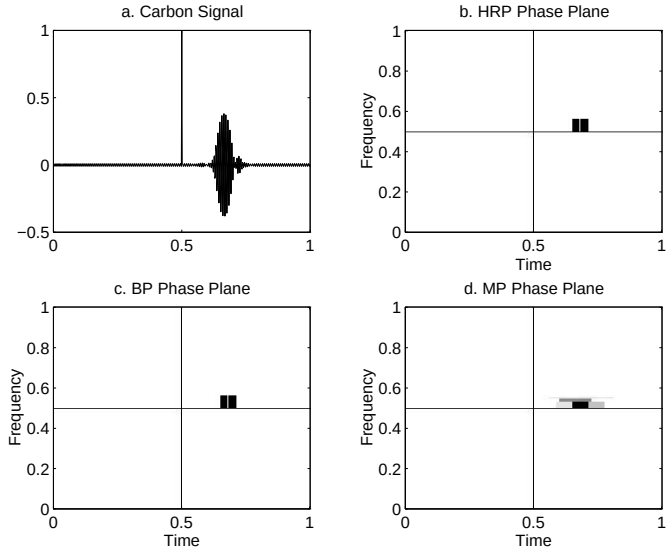

Figura 4.8: *Descomposición de una señal formada por cuatro elementos del diccionario. (a) Señal en el tiempo. (b) Descomposición con HRP. (c) Descomposición con BP. (d) Descomposición con MP. Figura obtenida mediante el toolbox* atomizer *de Matlab.*

átomos correlados provocan un error en una de las iteraciones tempranas evitando de esta forma la mejor descomposición.

Esta elevada resolución temporal se consigue sin incrementar el orden de complejidad del algoritmo [Gribonval96]. En realidad, para el sub-diccionario de átomos de alta resolución $\mathbf{D}_a$ se implementa un MP, pero el número de multiplicaciones varía para el sub-diccionario de átomos de baja resolución $\mathbf{D}_b$. Para éste, no se calcula directamente la correlación de los átomos con la señal, ni la actualización de correlaciones, sino que desde la primera iteración es necesario implementar la ecuación (4.27). Esta ecuación necesita tantas divisiones por átomo $\mathbf{g}_m \in \mathbf{D}_b$ como tamaño tenga el subconjunto $\mathbf{L}_m$, por lo que el número de operaciones depende de este valor.

El algoritmo HRP ha sido utilizado con éxito en la extracción de características, en la resolución de direcciones de llegada [Jaggi98], así como en audio con un diccionario compuesto de con átomos de Gabor [Gribonval96] .

4.2.3. Resultados

A continuación, se incluyen una serie de ejemplos tomados de [Chen95], en los que se pretende analizar el rendimiento de algunos métodos para la obtención de descomposiciones atómicas. En general, cabe decir que el rendimiento de cada método es muy dependiente de la aplicación y la señal de entrada a analizar. Sin embargo, cada método tiene una serie de problemas, que pueden aparecer para cualquier señal de entrada. La finalidad de este apartado es mostrar gráficamente los problemas comentados previamente en la explicación de cada método.

En primer lugar, se presenta la figura 4.9, que pretende mostrar los problemas que se producen con el método *matching pursuits* cuando la señal de entrada está formada por elementos del diccionario con alta correlación cruzada entre ellos. La señal de prueba en este caso está formada

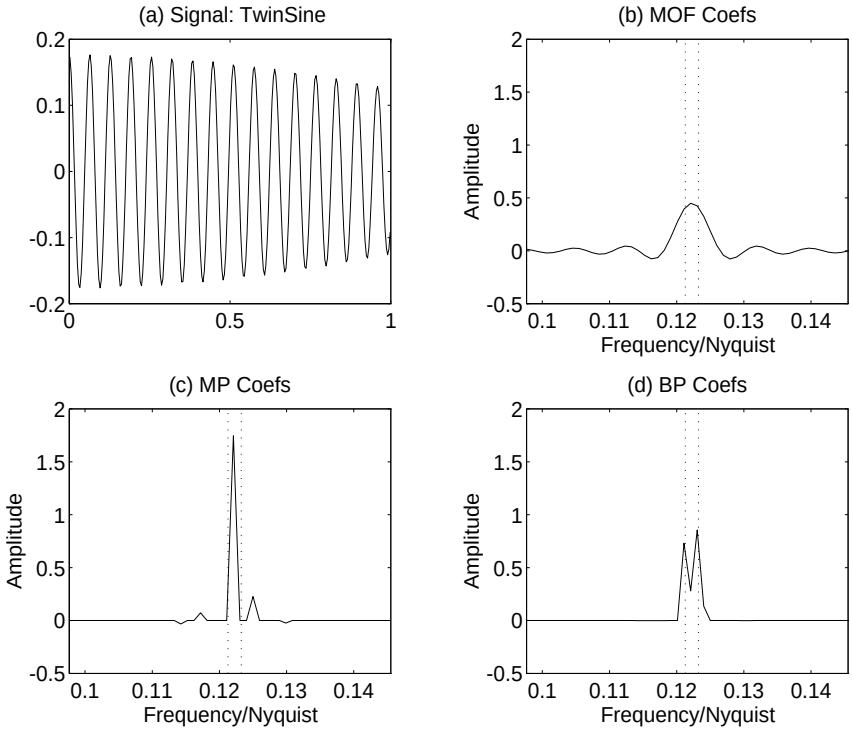

Figura 4.9: *Ejemplo de funcionamiento de diferentes métodos de obtención de descomposiciones atómicas con una señal formada por dos tonos muy próximos en frecuencia y un diccionario DST. Figura obtenida mediante el toolbox* atomizer *de Matlab.*

por dos senos muy próximos en frecuencia, y se analiza con un diccionario DST (*Discrete Sine Transform*) . Como se observa en la figura, el método MOF obtiene una representación muy poco compacta, puesto que la energía de la señal se reparte entre muchos elementos del diccionario. Además, el máximo de energía no está en los átomos que forman la señal sino en un máximo intermedio a ambos. Para el caso de *matching pursuits*, se produce un error en la primera iteración del diccionario, al extraer un átomo intermedio entre los dos que forman la señal. Se puede observar que este error no tiene solución, al tratarse de un método iterativo, y, en las iteraciones siguientes, el algoritmo simplemente trata de arreglar esta mala elección inicial. Por su parte, el algoritmo BP obtiene la descomposición ideal, pudiendo discriminar los dos átomos que forman la señal. Como conclusión, cabe decir que el método BP, al tratarse de un método paralelo, tiene una mayor resolución que los métodos iterativos y el método MOF, a costa de incrementar la complejidad computacional.

A continuación, se analiza la descomposición obtenida por los distintos métodos revisados, con señales sintéticas generadas, para comprobar la adaptación de cada método a sus características. En primer lugar, en la figura 4.10 se analiza una señal formada por una delta de Dirac, un tono y cuatro átomos wavelet packets. Para todos los métodos, se utiliza un diccionario wavelet packets. El plano de fase obtenido por el método MOF es muy difuso e incluye gran cantidad de elementos del diccionario. El método MP ofrece un buen resultado para la delta y el tono, pero comete errores al descomponer los cuatro átomos wavelet packets, al estar muy próximos en el plano tiempo-frecuencia. La descomposición obtenida bajo las siglas BOB (*Best Orthogonal Basis*) se

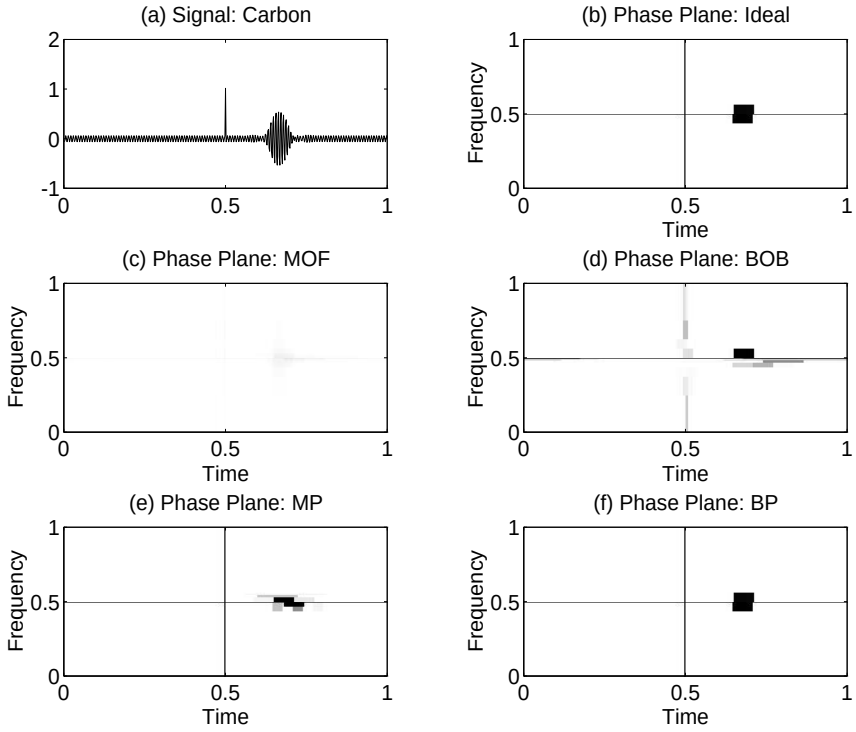

Figura 4.10: *Comparación del resultados de métodos para obtener descomposiciones con una señal formada por una delta de Dirac, un tono y cuatro funciones wavelet-packets. Se utiliza un diccionario wavelet packets. Figura obtenida mediante el toolbox* atomizer *de Matlab.*

consigue determinando la base wavelet packets ortogonal que obtiene la mejor descomposición bajo un criterio dado. En este caso, se puede observar como esta descomposición tiene problemas en distinguir correctamente la delta temporal del tono en el plano de fase, por lo que obtiene un resultado más pobre que MP. El mejor resultado, de nuevo, se obtiene con el método BP.

El siguiente ejemplo es el de la figura 4.11. En este caso, la señal no está formada por elementos del diccionario, ya que la señal se sintetiza mediante la suma de un tono puro y una señal FM obtenida mediante la modulación de un tono. El diccionario se forma a partir de un árbol cosine packets. Debido a que ahora el diccionario no puede representar de forma exacta la señal con unos pocos átomos, los planos de fase obtenidos no se corresponden en ningún caso con el plano de fase ideal. Pese a ello, se puede afirmar que el plano de fase del método BP es el más parecido al plano ideal, obteniendo un resultado similar a efectos prácticos los métodos MP y BOB, mientras que el método MOF vuelve a producir el resultado más pobre.

Como conclusión, cabe decir que, en general, el método BP obtiene una descomposición más apropiada, a costa de incrementar el coste computacional. El método MP obtendrá una descomposición mejor que la determinación de la mejor base ortogonal (BOB) cuando el diccionario sea altamente sobrecompleto, puesto que en este caso el método MP dispone de muchos átomos para realizar la descomposición, y el método BOB, al fin y al cabo, es la mejor transformada posible (que tiene tantas funciones base como longitud tenga la señal). El método MOF no tiene aplicación práctica, puesto que los resultados que se obtienen no son satisfactorios en ningún caso.

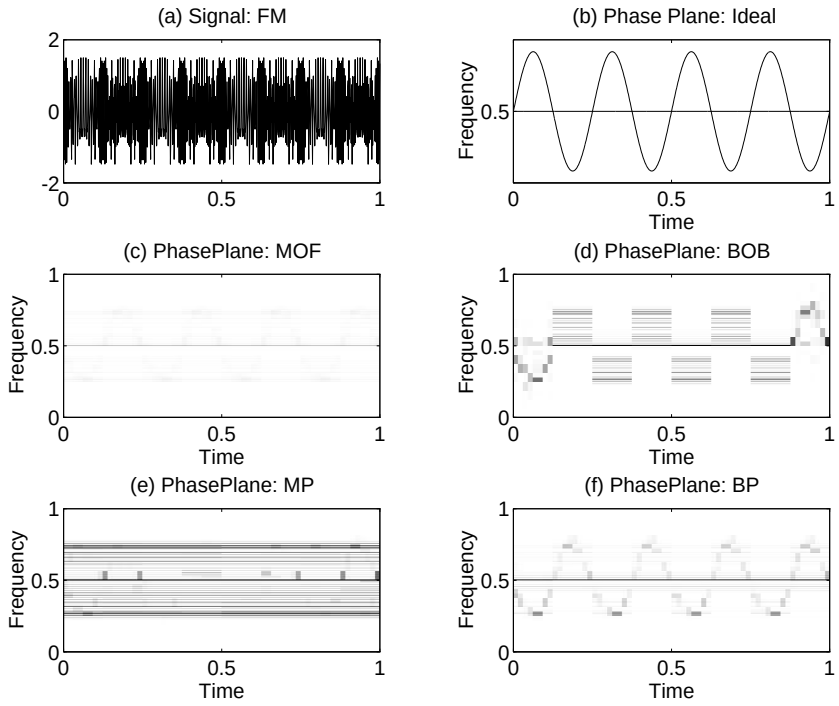

Figura 4.11: *Comparación del resultados de métodos para obtener descomposiciones con una señal formada un tono más una señal tonal modulada en FM. Se utiliza un diccionario cosine packets. Figura obtenida mediante el toolbox* atomizer *de Matlab.*

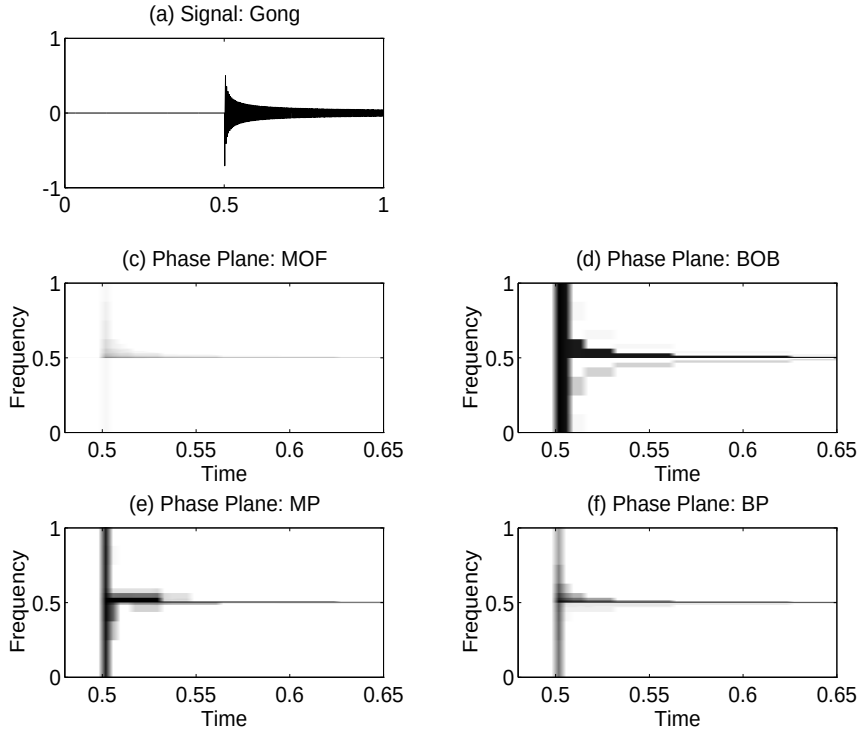

Figura 4.12: *Comparación del resultados de métodos para obtener descomposiciones con un transitorio de audio. Se utiliza un diccionario cosine packets. Figura obtenida mediante el toolbox* atomizer *de Matlab.*

Para finalizar los ejemplos de señales, se introduce la figura 4.12 donde se analiza una señal transitoria de audio con un diccionario de cosine-packets. El plano de fase ideal de esta señal debe reflejar altas frecuencias en el momento del golpe del instrumento y, después, la frecuencia debe acercarse cada vez más a la frecuencia de resonancia producida. Como se observa en la figura, este plano de fase se obtiene con todos los métodos, sin que haya una diferencia apreciable entre ellos (salvo para MOF). Quizás, la descomposición con BP proporcione un resultado más claro, aunque las diferencias son mínimas. Así pues, para señales prácticas, que no están formadas por elementos del diccionario, y que no tienen un plano tiempo frecuencia conocido, los resultados obtenidos son similares para todos los métodos (salvo para MOF).

Por último, es interesante incluir un ejemplo práctico de aplicación para las descomposiciones atómicas. Para varios métodos de descomposición como BP o MP, se puede obtener una aproximación de la señal en pocas iteraciones de los algoritmos de cálculo. Por lo tanto, una aplicación ideal para probar la validez de estos métodos es la eliminación de ruido o *denoising*. En el caso de *matching pursuits*, al utilizar la correlación como herramienta de cálculo, la potencialidad para eliminar ruido blanco de la señal (incorrelado con la misma y con los átomos del diccionario) es enorme. En el caso de *basis pursuits*, si se utiliza como algoritmo de cálculo *interior point*, se puede detener el algoritmo en una iteración temprana con una aproximación basada en unos pocos átomos de la señal. En general, para todas las descomposiciones, la eliminación de ruido se realiza mediante umbralización, ya sea de la transformada obtenida mediante BOB, de la descomposición por el método MOF, o deteniendo el cálculo de la descomposición en una iteración dada en MP o BP con *interior point*.

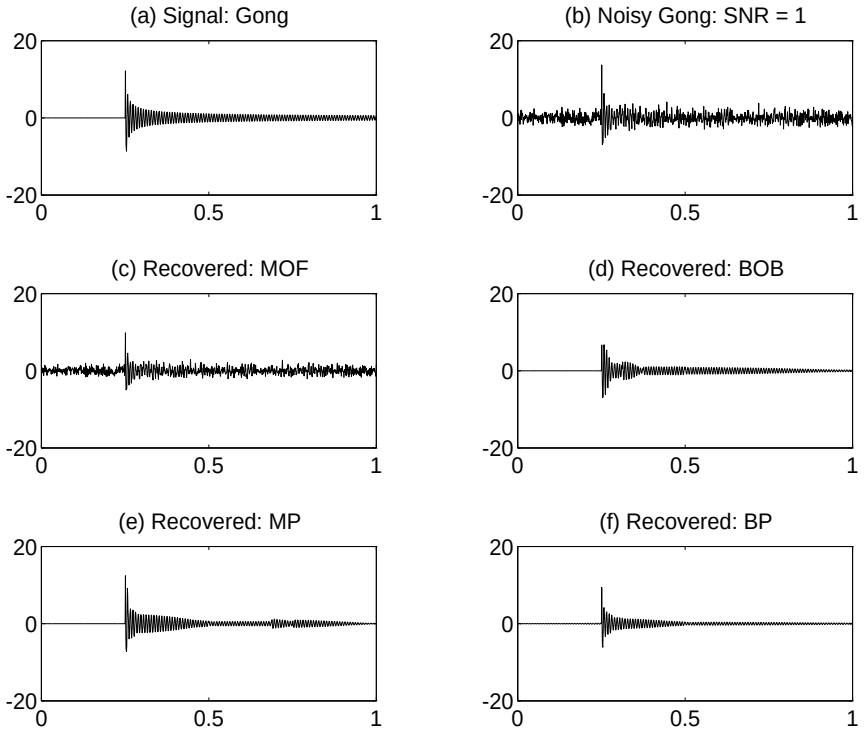

Figura 4.13: *Comparación del resultados de métodos de descomposiciones para eliminación de ruido en un transitorio de audio. Figura obtenida mediante el toolbox* atomizer *de Matlab.*

En la figura 4.13 se dibuja la señal resultado para las diferentes formas de obtener una descomposición. Como se observa en la parte de señal donde está el transitorio, los resultados son similares para BP y MP, y algo peores para BOB, siendo realmente malos para MOF.

Para aplicaciones prácticas, los diferentes métodos obtienen resultados similares. No obstante, conforme más complejo es el método, tiende a obtener mejores resultados. En la literatura ha sido muy utilizada la descomposición BOB (que estrictamente hablando no es una descomposición atómica), por ejemplo en audio [Ruiz01]. Sin embargo, dentro de una complejidad razonable, está siendo utilizado cada vez más el método MP [Heusdens02] [Verma99], que obtiene una descomposición atómica y, por tanto, puede ser aplicado para obtener codificadores paramétricos de audio.

4.3. Tipos de diccionarios tiempo-frecuencia

Para el desarrollo de un modelo general de descomposición de señal, los átomos han de ser elegidos de forma que se correspondan con las características básicas de la señal. Este enfoque se recomienda especialmente para aplicaciones de análisis y codificación de señal, porque cada átomo puede describir un comportamiento concreto de la señal de entrada. Además, si el diseño del diccionario es paramétrico, en el sentido de que cada átomo esté definido por parámetros con significado físico, tales como localización temporal, frecuencia de modulación o escala, la relación entre estos parámetros y las características propias de la señal puede ser muy directa.

Caben distinguir dos enfoques en el diseño del diccionario. Por un lado, los primeros trabajos en descomposiciones atómicas siguieron el camino de utilizar un gran diccionario generalista que incluyera una gran variedad de comportamientos en la definición paramétrica de los átomos [Mallat93]. Sin embargo, esta aproximación no es óptima desde el punto de vista de la complejidad por el gran tamaño del diccionario. Como consecuencia, se suele diseñar el diccionario en función de la aplicación para la que se utilice la descomposición. Por ejemplo, para realizar un modelado sinusoidal parece lógico implementar un diccionario compuesto de exponenciales complejas, sin considerar funciones transitorias dentro diccionario. A continuación, se revisan las características de los diccionarios más utilizados en la bibliografía para el cálculo de descomposiciones atómicas, así como sus principales utilidades.

4.3.1. Átomos de Gabor

Este tipo de átomos se utiliza cuando se diseña un diccionario de grandes dimensiones para proporcionar una representación tiempo-frecuencia de la señal, siendo la representación resultante paramétrica y compacta [Mallat93]. Los átomos de Gabor [Gabor46] son funciones con una buena localización tiempo-frecuencia, que permiten modelar un gran rango de comportamientos de la señal a analizar. Debido a esta propiedad, los átomos de Gabor se han convertido en un clásico en la bibliografía sobre el tema, siendo su uso recurrente en descomposiciones de carácter general.

En tiempo continuo, los átomos de Gabor se obtienen, a partir de una ventana $g(t)$, realizando una modulación en frecuencia y un desplazamiento y escalado en el tiempo:

$$g_{\{s,\omega,\tau\}}(t) = \frac{1}{\sqrt{s}} g\left(\frac{t-\tau}{s}\right) e^{j\omega(t-\tau)} \tag{4.31}$$

Esta definición se puede extender a tiempo discreto muestreando los átomos. Una forma de escribir este muestreo es la siguiente:

$$g_{\{s,\omega,\tau\}}[n] = f_s[n-\tau] e^{j\omega(n-\tau)} \tag{4.32}$$

donde $f_s[n]$ es una ventana de energía unidad que se puede escalar mediante el parámetro s.

Como se puede observar, aparte de normalizar la energía de cada átomo, un diccionario formado por átomos de Gabor se indexa a partir de los parámetros $\{s, \omega, \tau\}$. Esta estructura permite una descripción muy simple de cada átomo, siendo muy valiosa en aplicaciones de análisis o compresión. Un diccionario compuesto por átomos de Gabor suele ser altamente sobre-completo e incluir bases de Fourier (al variar la frecuencia) y wavelet (al variar la escala). Sin embargo, este tipo de diccionario tiene unos importantes inconvenientes. Por un lado, debido al gran tamaño necesario para abarcar un conjunto suficientemente representativo de comportamientos tiempo-frecuencia, la complejidad requerida para el cálculo de la descomposición es demasiado elevada para la mayoría de las aplicaciones de compresión, aunque se hayan ideado algunos algoritmos para reducir la complejidad [Goodwin97]. Además, el resultado de la descomposición es altamente dependiente de la ventana utilizada. De forma general, se utilizan ventanas simétricas. La figura 4.14 representa varios átomos a diferente frecuencia y escala para este caso concreto de ventana simétrica.

Sin embargo, en aplicaciones donde la señal tiene un comportamiento transitorio, los átomos de Gabor pueden no ser la mejor elección. En la figura 4.15 se aprecia el pre-eco producido por

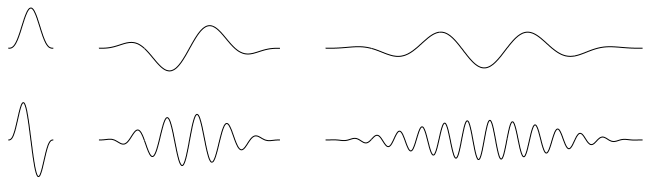

Figura 4.14: *Átomos de Gabor con ventana simétrica variando la frecuencia de modulación y la escala de la ventana.*

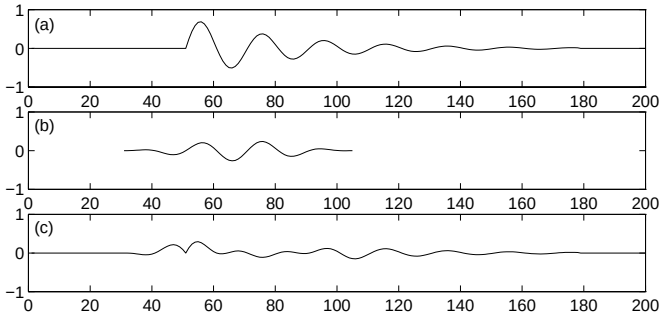

Figura 4.15: *Representación de un efecto de pre-eco producido al utilizar átomos de Gabor simétricos. (a) Señal formada por una exponencial amortiguada. (b) Primer átomo de Gabor elegido mediante* matching pursuits. *(c) Residuo tras la primera iteración donde se aprecia el efecto de pre-eco.*

la elección de una ventana simétrica en una señal formada por una exponencial amortiguada.

Una solución a este problema puede consistir en el empleo del algoritmo *high resolution pursuits* para el cálculo de la descomposición, aunque para señales transitorias puede haber otros diccionarios más apropiados, como por ejemplo el formado por sinusoides amortiguadas.

Este problema, surgido de la imposibilidad de un diccionario de representar correctamente aquellas señales cuyo comportamiento no se adapta a los átomos que lo forman, puede aparecer en función de las características de la señal. Así, para el caso de la señal de audio o de la señal radar, es común encontrar ejemplos donde la frecuencia de la señal cambia con el tiempo, algo no contemplado en la definición de los átomos de Gabor. Esta modulación en frecuencia o *chirp* se puede incluir implementando un diccionario aún más amplio [Gribonval01]. Sin embargo, este enfoque, basado en generalizar el diccionario para responder a cualquier comportamiento de la señal de entrada, no parece la solución más idónea cuando la complejidad de la descomposición se incrementa exponencialmente con el tamaño del diccionario. Al contrario, para poder realizar descomposiciones acordes al problema a solucionar, se diseñan diccionarios a medida, donde se mantienen sólo las propiedades a analizar, reduciendo así el número de átomos y por tanto la complejidad. A continuación, se presentan una serie de diccionarios menos generalistas, adaptados a aplicaciones concretas, a partir de los cuales se analizarán los problemas que aparecen en la descomposición de la señal de audio.

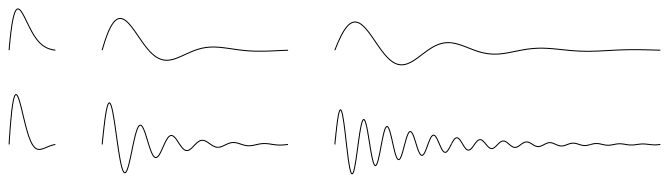

Figura 4.16: *Átomos de sinusoides amortiguadas variando la frecuencia de modulación y el factor de amortiguamiento.*

4.3.2. Sinusoides amortiguadas

La aparición de oscilaciones con caída exponencial en muchas señales naturales justifica la consideración de tonos que tienen una caída exponencial para el diseño del diccionario. Al fin y al cabo, una señal de estas características es la respuesta al impulso de un filtro con un polo complejo, siendo esta propiedad interesante cuando se intentan modelizar señales producidas por sistemas lineales.

Un diccionario formado por estas señales representa de forma más eficaz los transitorios presentes en señales reales que un diccionario formado por átomos de Gabor con ventanas simétricas [Goodwin97]. Un diccionario formado por sinusoides amortiguadas se indexa al igual que un diccionario formado por átomos de Gabor con tres parámetros, que ahora son: el factor de amortiguamiento a, la frecuencia de modulación ω y el tiempo de comienzo τ:

$$g_{\{a,\omega,\tau\}}[n] = S_a a^{(n-\tau)} e^{j\omega(n-\tau)} u[n-\tau] \tag{4.33}$$

donde el factor S_a se incluye para conseguir norma unidad. Ejemplos de átomos de este tipo se dibujan en la figura 4.16. Lógicamente, este tipo de átomos se puede definir también a partir de átomos de Gabor con una ventana basada en una exponencial multiplicada por la función impulso unidad.

En la literatura han aparecido diversos enfoques que utilizan átomos tiempo-frecuencia con comportamiento exponencial. En [Friedlander95], las sinusoides amortiguadas se utilizan para proporcionar una representación tiempo-frecuencia a partir de la cual los transitorios de la señal se pueden identificar fácilmente. Sin embargo, en esta referencia se supone un cierto conocimiento a priori del factor de amortiguamiento, que es razonable para aplicaciones de detección, pero inapropiado para obtener descomposiciones de señales arbitrarias. En cualquier caso, un diccionario con sinusoides amortiguadas obtiene una descomposición adecuada para señales transitorias [Goodwin97b]. En el caso concreto de la señal de audio, cuando se desea aplicar una descomposición atómica, este diccionario obtiene una descomposición adecuada para representar la parte transitoria [Nieuwenhuijse98]. El precio a pagar es una complejidad muy alta para el modelado de los transitorios de audio, lo que ha hecho que su uso sea reducido en codificadores paramétricos de audio.

4.3.3. Exponenciales complejas

Cuando la descomposición atómica se va a realizar sobre una señal muy tonal, como una señal armónica de audio, los átomos a extraer deben estar basados en sinusoides. En el caso

particular de una señal real, el conjunto de átomos debería estar formado por un conjunto de tonos de diferente frecuencia ω y fase ϕ:

$$g_{\{\omega,\phi\}}[n] = S_c cos\left(\omega n + \phi\right) \tag{4.34}$$

donde S_c es una constante para obtener potencia unidad. Con estos átomos, el tamaño del diccionario viene determinado por el número de frecuencias y fases, siendo el tamaño total la multiplicación de ambos valores. Sin embargo, haciendo uso de la matemática compleja, el tamaño del diccionario se puede reducir significativamente. Así, definiendo los átomos como exponenciales complejas:

$$g_{\omega}[n] = S_e e^{j\omega n} \tag{4.35}$$

no es necesario incluir la fase en la definición de los átomos, puesto que ésta se calcula por medio de la matemática discreta. Por ejemplo, si se aplica *matching pursuits*, la correlación con números reales es un valor real de amplitud, mientras que para números complejos, la correlación de la señal con cada átomo es ahora un valor complejo con amplitud y fase. Con esta definición de los átomos como exponenciales complejas, se reduce por tanto el tamaño del diccionario a sólo el número de frecuencias que necesite la descomposición atómica.

En el caso particular de emplear *matching pursuits* para el cálculo de la descomposición atómica, es posible realizar una modificación del algoritmo que permite un funcionamiento apropiado con átomos complejos y señales reales. Esta modificación del algoritmo se basa en extraer en cada iteración un subespacio de señal, por lo que se conoce en la bibliografía como *subspace pursuits* [Verma99] [Goodwin97]. Un subespacio de señal se define como una suma ponderada de varios átomos, siendo estos subespacios la señal que se extrae en cada iteración. En el caso de exponenciales complejas, el subespacio de señal está formado por la suma de cada átomo y su complejo conjugado. De esta forma, cada subespacio es real y, como estos subespacios son los que se extraen de la señal, el residuo final también lo es. Esta definición de subespacios para átomos exponenciales complejos se denomina en inglés *conjugate subspaces*. Para esta situación, en cada iteración i, se extrae un átomo $\mathbf{g}_{\omega(i)}$ con su coeficiente (o peso) asociado $\alpha_{\omega(i)}$ y su conjugado,

$$\mathbf{r}^i = \begin{cases} \mathbf{x} & i = 0 \\ \mathbf{r}^{i-1} - (\alpha_{\omega(i)}\mathbf{g}_{\omega(i)} + \alpha^*_{\omega(i)}\mathbf{g}^*_{\omega(i)}) & i > 0 \end{cases} \tag{4.36}$$

donde el coeficiente $\alpha_{\omega(i)} = |\alpha_{\omega(i)}|e^{j\phi}$ es un valor complejo con módulo y fase, pudiéndose escribir el residuo en función del coseno para el caso de átomos exponenciales complejos como,

$$\mathbf{r}^i = \begin{cases} \mathbf{x} & i = 0 \\ \mathbf{r}^{i-1} - 2S_e|\alpha_{\omega(i)}|cos(\omega n + \phi) & i > 0 \end{cases} \tag{4.37}$$

Aunque ahora la obtención de los coeficientes no se basa simplemente en la correlación de la señal con los átomos, si depende muy directamente de este valor [Goodwin97]. Además, la correlación de una señal discreta con un conjunto de exponenciales complejas es el cálculo de la Transformada Discreta de Fourier de la señal.

Las exponenciales complejas han sido utilizadas como átomos en el cálculo de descomposiciones atómicas con la finalidad de aplicar un modelado sinusoidal. Este modelo de señal se ha

aplicado para voz y audio con generalidad [Mcaulay86]. Otra ventaja adicional de este enfoque consiste en la introducción de información psicoacústica para el cálculo de los coeficientes asociados a cada exponencial. No se extrae el átomo que minimiza la energía del residuo, si no aquel que minimiza la importancia perceptual del mismo. Se han propuestos varios enfoques en la bibliografía para este problema [Verma99b] [Heusdens02].

4.3.4. Diccionarios basados en transformadas

Como en el caso de las exponenciales complejas, es posible definir diccionarios basados en transformadas y bancos de filtros. Como se ha visto en el apartado anterior, es posible definir un diccionario de exponenciales complejas y aplicar la Transformada Discreta de Fourier (*Discrete Fourier Transform, DFT*) para *matching pursuits*. Esta propiedad se deriva del hecho de que tanto *matching pursuits*, como otros métodos para calcular descomposiciones atómicas, utilizan como medida la correlación entre los átomos y la señal. Esta correlación se calcula directamente al aplicar una transformada o un banco de filtros, si se definen los átomos a partir de la respuesta del banco de filtros de síntesis o de las funciones base de la expresión de la transformada inversa.

La transformada o banco de filtros a utilizar depende en gran medida de la aplicación a desarrollar o de la señal de entrada. En este sentido, cuando se busca implementar un modelado sinusoidal mediante una descomposición atómica parece lógico utilizar la Transformada Discreta de Fourier como herramienta de cálculo de las correlaciones, lo que da lugar a un diccionario de exponenciales complejas. También es posible utilizar otras transformadas, como la *cosine packets*, *sine packets*, Transformada Discreta del Coseno o del Seno, basadas en elementos sinusoidales, que tienen la posibilidad (al igual que la DFT) de definir un número de coeficientes de la transformada superior a la longitud de la señal de entrada para poder obtener un diccionario sobrecompleto. En el caso de la DFT, esto se consigue rellenando con ceros la señal de entrada.

Cuando se desea implementar un modelado de transitorios, las transformadas anteriores no son las más idóneas. En su lugar, es mejor utilizar transformadas como la Transformada Wavelet o Wavelet-Packets, que obtienen mejores resultados cuando se pretende representar componentes transitorias presentes en la señal de audio [Ruiz01]. Sin embargo, para cumplir el requisito de que el diccionario sea sobrecompleto, es necesario la elección de la Transformada Wavelet-Packets (*Wavelet-Packets Transform, WPT*) . El uso de una WPT para implementar un modelado de transitorios mediante una descomposición atómica es muy prometedor [Vera04a]. En el caso de un diccionario wavelet packets, los átomos que lo forman son todas aquellas funciones del árbol de descomposición hasta el nivel de descomposición J, siendo este valor el número de veces que es sobrecompleto el diccionario. Los átomos de este diccionario se identifican a partir de tres índices $\{s, p, r\}$, que indican la subbanda en un nivel de descomposición dado, la profundidad de descomposición y el retardo dentro de cada subbanda, respectivamente. Los átomos se pueden definir como,

$$g_{\{s,p,r\}}[n] = g_{\{s,p\}}[n - 2^p r] \tag{4.38}$$

donde la secuencia $g_{\{s,p\}}[n]$ es la versión en el tiempo de la función en el dominio z, $G_{\{s,p\}}(z)$. Esta función se puede calcular directamente a partir de las funciones de transferencia de los filtros de síntesis paso bajo y paso alto, $G_0(z)$ y $G_1(z)$, respectivamente, de la transformada WPT. Estos

filtros son los que implementan la transformada WPT inversa. Resumiendo, la función $G_{\{s,p\}}(z)$ se puede expresar como [Vera04a]:

$$G_{\{s,p\}}(z) = \prod_{d=0}^{p-1} G_{((\lfloor s/2^d \rfloor))_2}(z^{2^d}) \tag{4.39}$$

donde $((k))_L$ representa (k modulo L). Como se puede observar, una vez que se decide la transformada a utilizar, los átomos de la descomposición son las respuestas al impulso de cada coeficiente de la transformada. De esta forma, el cálculo de la correlación se simplifica a calcular la transformada directa.

Para el caso del método *matching pursuits*, la ventaja adicional de utilizar un diccionario estructurado en una transformada es la obtención sencilla de las correlaciones cruzadas entre átomos necesarias para la actualización de las mismas. Por ejemplo, en el caso de diccionario wavelet packets, la memoria para guardar las correlaciones cruzadas se reduce a la necesaria para almacenar las respuestas al impulso de cada función base o coeficiente [Vera04a]. Debido a esta característica, los diccionarios basados en transformadas necesitan menos requerimientos de memoria que otros diccionarios previamente revisados, como los átomos de Gabor o las sinusoides amortiguadas. En general, los diccionarios basados en una transformada precisan un número de átomos reducido. En el caso de la WPT, dicho número viene determinado por la profundidad de la transformada. Además, estrictamente hablando, no se trata de diccionarios paramétricos, en el sentido de que los parámetros del diccionario no tienen sentido físico en relación a la señal a analizar.

4.3.5. Diccionarios mixtos

Una forma de conjugar la adaptación de los diccionarios revisados a las características de la señal de entrada es la utilización de diccionarios mixtos. En este sentido, la incapacidad de los átomos de Gabor de representar transitorios de señal se soluciona incluyendo en la definición del diccionario un conjunto de átomos con las características de las sinusoides amortiguadas. Esta unión de diccionarios se ha utilizado con el objetivo último de definir un diccionario generalista que incluya el máximo de comportamientos tiempo-frecuencia para adaptarse a la señal de entrada [Goodwin97] [Gribonval01]. Siempre que se propone el uso de un diccionario mixto de este tipo es necesario hacer un estudio para reducir al máximo los requisitos de memoria y complejidad de la descomposición atómica. Aún así, el uso de un diccionario de este tipo es muy restringido, debido a la alta complejidad asociada al cálculo de la descomposición.

Un ejemplo de funcionamiento para comprobar las ventajas de utilizar un diccionario mixto aparece en la figura 4.17. La señal de prueba está formada por la suma de un seno más dos impulsos unidad desplazados en el tiempo. Como se puede comprobar, el uso independiente de los diccionarios individuales con el algoritmo BP no produce un resultado compacto, puesto que los átomos no están adaptados a todas las características de la señal. En cambio, con un diccionario mixto (o *merge* en inglés) se obtiene un resultado altamente satisfactorio, porque recoge en tres coeficientes las propiedades de la señal de entrada.

Para el caso de la señal de audio que incluye un comportamiento muy tonal con zonas transitorias, el uso de un diccionario mixto puede ser una herramienta de gran potencia para separar en análisis la parte tonal de la transitoria. Sin embargo, la complejidad no puede ser muy alta si

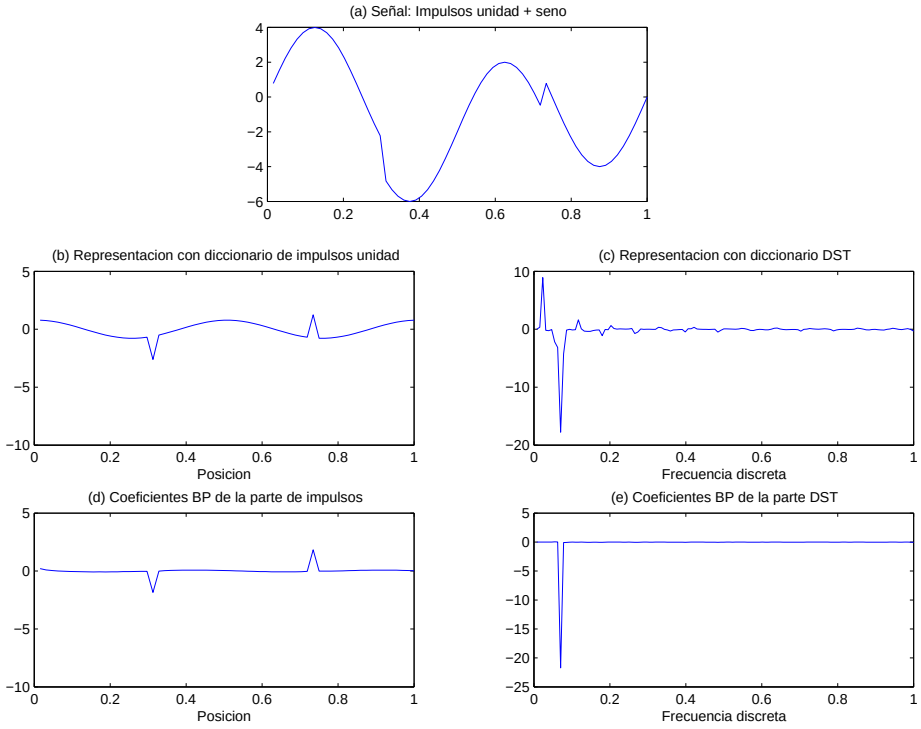

Figura 4.17: *Ejemplo de uso de un diccionario mixto. (a) Señal formada por un seno más dos impulsos unidad retardados en el tiempo . (b) Coeficientes obtenidos mediante BP con un diccionario de impulsos unidad. (c) Coeficientes con BP y un diccionario DST (Discrete Sine Transform). (d) Coeficientes de la parte de impulsos unidad del diccionario mixto (impulsos unidad más DST) con BP. (e) Coeficientes de la parte DST del diccionario mixto (impulsos unidad más DST) con BP.*

se pretende realizar una descomposición con posible aplicación práctica. Una idea desarrollada con cierto éxito es la de utilizar un diccionario mixto formado por exponenciales complejas, que se adapten a la parte tonal de la señal, y átomos derivados de la transformada wavelet packets, apropiados para capturar la parte transitoria del audio [Vera05a]. El principal problema a solventar, si se utiliza el algoritmo *matching pursuits*, es la actualización de las correlaciones, que exige almacenar en memoria las correlaciones cruzadas entre átomos sinusoidales y wavelets packets. En este caso concreto, sólo es preciso guardar la transformada discreta de Fourier de las funciones wavelets packets [Vera05a].

En general, para el caso de los algoritmos iterativos (por ejemplo *matching pursuits*), el uso de un diccionario mixto obtiene mejores resultados que la aplicación en serie de diccionarios individuales. La causa de esta afirmación hay que buscarla en que si se aplican descomposiciones en serie surge el problema de cuándo parar una descomposición para empezar la siguiente con el residuo resultante. Si se realizan la descomposición en base a pocos átomos extraídos, es muy posible que no se hayan extraído todos los átomos que modelen el comportamiento de la señal adaptado al primer diccionario. Al contrario, si se permite la extracción de demasiados átomos en la primera descomposición, se puede modificar el comportamiento de la señal en relación al siguiente diccionario. Al fin y al cabo, el problema de determinar la parada del algoritmo se soluciona con una solución de compromiso. Sin embargo, con el uso de un diccionario mixto este problema no se produce, puesto que se extrae en cada iteración el átomo óptimo, calculado teniendo en cuanta ambos diccionarios. Como contrapartida, la complejidad de la descomposición crece, sobre todo con la inclusión dentro del diccionario mixto de átomos de diferente naturaleza.

Parte II

Desarrollo y Metodología de la Investigación

Capítulo 5

Modelado sinusoidal

A partir de la revisión realizada anteriormente acerca del modelado sinusoidal, se llega a la conclusión de que hay una gran cantidad de posibles opciones a la hora de mejorar los resultados de este modelo implementado mediante una simple detección de picos espectrales en la DFT. A la hora de elegir la opción más apropiada es muy importante tener en cuenta las siguientes consideraciones:

Tiempo computacional Cuando el funcionamiento en tiempo real es un requisito indispensable a la hora de implementar el modelado tonal, es necesario evitar algoritmos con carácter iterativo [Myburg04]. Bajo esta premisa, los métodos de cálculo disponibles son aquellos que optimizan los parámetros sinusoidales de manera global. Para este grupo de algoritmos, se utiliza como primera aproximación los valores calculados mediante la DFT y la detección de picos espectrales para, posteriormente, utilizar algún tipo de método numérico para minimizar las desviaciones que se producen en el cómputo de los parámetros sinusoidales. Los métodos más utilizados en codificación paramétrica de audio son la estimación por máxima semejanza propuesta en [Thomson82] y utilizada en el codificador HILN [Purnhagen02] y en los codificadores híbridos de Ali [Ali95] y Levine [Levine98], y la minimización por mínimos cuadrados propuesta en [George87] y usada en la versión escalable del codificador PPC [Myburg04].

Inclusión de información perceptual Los mejores resultados prácticos a la hora de implementar un modelo sinusoidal se consiguen mediante el algoritmo *matching pursuits* [Myburg04]. El precio a pagar es el incremento de la complejidad computacional debido al uso de un algoritmo de carácter iterativo. Sin embargo, el uso de *matching pursuits* aporta como ventaja adicional la posible modificación del algoritmo para incluir información psicoacústica en la selección del tono más importante en cada iteración. Así, por definición [Mallat93], el método MP elige en cada iteración el elemento del diccionario que extrae mayor energía del residuo actual. Esta medida de energía, que se calcula mediante la correlación, se puede modificar [Verma99b] [Heusdens02b] para extraer el átomo perceptualmente más importante con la simple introducción en las ecuaciones del umbral de enmascaramiento calculado para los tonos. Con esta modificación, el algoritmo MP tiene una gran potencialidad como herramienta para calcular el modelo tonal en aplicaciones de codificación de audio, de tal forma que en este apartado se revisará con detalle la forma de implementar

un método *matching pursuits* perceptual.

5.1. Implementación mediante *matching pursuits*

Como se comentó en el apartado 4.3.3, para la implementación de un modelo sinusoidal mediante el método MP, es una ventaja disponer de un diccionario de exponenciales complejas, aunque la señal sea real, puesto que se limita el tamaño del diccionario al número de frecuencias a buscar en la señal, mientras que la fase se calcula como parte de la correlación en las operaciones complejas. Eso sí, para trabajar con señales reales, es necesario utilizar subespacios de señal formados por cada átomo y su conjugado al objeto de extraer en cada iteración sólo señales reales. Esta pequeña modificación del algoritmo *matching pursuit* no cambia sustancialmente su funcionamiento.

Para implementar el modelado sinusoidal de señales de longitud finita, el diccionario sobrecompleto D debe estar compuesto de funciones sinusoidales enventanadas que puedan variar en frecuencia y fase. Como se ha comentado, el empleo de funciones exponenciales complejas reduce la complejidad computacional del modelo tonal en comparación con el empleo de funciones senoidales reales. Como se muestra a continuación, la proyección de cada función exponencial compleja contiene la información de fase.

El diccionario está formado, por tanto, de un conjunto de funciones exponenciales complejas enventanadas que se puede definir de la forma,

$$g_k[n] = S_w w[n] e^{j\frac{2\pi k}{2L}n}, \quad k = 0, \ldots, L, \quad n = 0, \ldots, N-1 \tag{5.1}$$

donde la constante S_w se elige para obtener átomos de norma unidad, $w[n]$ es la ventana de análisis de longitud N y $L+1$ el número de frecuencias dentro del diccionario.

En cada iteración, el residuo se calcula a partir de la expresión (en notación matricial),

$$\mathbf{r}^i = \begin{cases} \mathbf{x} & i = 0 \\ \mathbf{r}^{i-1} - (\alpha_{k(i)}\mathbf{g}_{k(i)} + \alpha^*_{k(i)}\mathbf{g}^*_{k(i)}) & i > 0 \end{cases} \tag{5.2}$$

donde el coeficiente $\alpha_{k(i)} = |\alpha_{k(i)}|e^{j\phi}$ es un valor complejo con módulo y fase, pudiéndose escribir en función del coseno como,

$$\mathbf{r}^i = \begin{cases} \mathbf{x} & i = 0 \\ \mathbf{r}^{i-1} - 2S_w\mathbf{w}|\alpha_{k(i)}|cos(\frac{2\pi k}{2L}n + \phi) & i > 0 \end{cases} \tag{5.3}$$

Para conocer el átomo óptimo y su coeficiente asociado, se minimiza la energía del residuo en cada iteración,

$$\min_{\mathbf{g}_k \in \mathbf{D}} ||\mathbf{r}^i||^2 \quad para \quad \mathbf{r}^i = \mathbf{r}^{i-1} - 2Re\{\boldsymbol{\alpha}^i_k \mathbf{g}_k\} \tag{5.4}$$

En cada iteración se elige el peso que minimiza la norma l^2 de $\mathbf{r}^i$, que se corresponde con el coeficiente del átomo óptimo $\alpha_{i(k)}$. Introduciendo el valor de $\mathbf{r}^i$ y minimizando la función, se obtiene que lo anterior equivale a maximizar

$$\max_{\mathbf{g}_k \in D} \left| 2Re\{(\boldsymbol{\alpha}^i_k)^* < \mathbf{r}^{i-1}, \mathbf{g}_k >\} - |\boldsymbol{\alpha}^i_k|^2 \|\mathbf{g}_k\|^2 - Re\{(\boldsymbol{\alpha}^i_k)^2 < \mathbf{g}_k, \mathbf{g}^*_k >\} \right| \tag{5.5}$$

Maximizando esta función, la solución final se puede expresar de la forma [Goodwin97],

$$\max_{\mathbf{g}_k \in \mathbf{D}} ||\boldsymbol{\alpha}_k^i||^2 \quad para \quad \boldsymbol{\alpha}_k^i = \frac{< \mathbf{r}^{i-1}, \mathbf{g}_k > - < \mathbf{r}^{i-1}, \mathbf{g}_k >^* < \mathbf{g}_k^*, \mathbf{g}_k >}{1 - | < \mathbf{g}_k^*, \mathbf{g}_k > |^2} \tag{5.6}$$

Se puede comprobar que el cálculo de los pesos de cada átomo cumple el principio de ortogonalidad, y que ahora el residuo en la iteración i, $\mathbf{r}^i$, es ortogonal al átomo óptimo $g_{k(i)}$ y a su conjugado,

$$\begin{aligned} < \mathbf{r}^i, \mathbf{g}_{k(i)} > &= 0 \\ < \mathbf{r}^i, \mathbf{g}_{k(i)}^* > &= 0 \end{aligned} \tag{5.7}$$

De la misma forma que para el algoritmo *matching pursuits* con un diccionario simple, para el caso de subespacios de complejos conjugados, es también posible la obtención de una ecuación para actualizar las correlaciones de forma rápida,

$$< \mathbf{r}^i, \mathbf{g}_k > = < \mathbf{r}^{i-1}, \mathbf{g}_k > - \boldsymbol{\alpha}_{k(i)} < \mathbf{g}_{k(i)}, \mathbf{g}_k > - \boldsymbol{\alpha}_{k(i)}^* < \mathbf{g}_{k(i)}^*, \mathbf{g}_k > \tag{5.8}$$

5.1.1. Implementación eficiente

Gracias al empleo de átomos complejos conjugados, las correlaciones necesarias se pueden calcular de manera eficiente aplicando la transformada rápida de Fourier (FFT, *Fast Fourier Transform*). Esto es posible puesto que las correlaciones entre la señal $x[n]$ y los átomos $g_k[n]$ son simplemente una Transformada Discreta de Fourier (DFT, *Discrete Fourier Transform*),

$$< x[n], g_k[n] > = S_w \sum_{n=0}^{N-1} x_w[n] e^{-j\frac{2\pi k}{2L}n} = S_w \cdot \ X_w[k] \tag{5.9}$$

donde $x_w[n] = x[n] \cdot w[n]$ es la señal de entrada multiplicada por la ventana, $X_w[k]$ es la DFT de longitud $2L$ de la señal de entrada enventanada y $L > N$, siempre que se quiera tener un diccionario sobre-completo. Para calcular las correlaciones iniciales, como se expresa en (5.9), es necesario, por tanto, aplicar la FFT a la señal de entrada enventanada, rellenando con ceros hasta llegar a una longitud $2L$.

La misma consideración se puede hacer a la hora de pre-calcular las correlaciones cruzadas entre los átomos del diccionario,

$$\begin{aligned} < g_{k(i)}[n], g_k[n] > &= |S_w|^2 \textstyle\sum_{n=0}^{N-1} u[n] e^{-j\frac{2\pi(k-k(i))}{2L}n} \\ &= |S_w|^2 U[((k-k(i)))_{2L}] \end{aligned} \tag{5.10}$$

$$\begin{aligned} < g_{k(i)}^*[n], g_k[n] > &= |S_w|^2 \textstyle\sum_{n=0}^{N-1} u[n] e^{-j\frac{2\pi(k+k(i))}{2L}n} \\ &= |S_w|^2 U[((k+k(i)))_{2L}] \end{aligned} \tag{5.11}$$

donde $u[n] = |w[n]|^2$ and $U[k]$ la DFT de longitud $2L$ de $u[n]$ y $()_{2L}$ denota modulo $2L$. Al ser $u[n]$ una señal real basta con almacenar $L+1$ valores de la transformada $U[k]$ para aplicar la actualización de correlaciones.

A partir de las ecuaciones (5.10) y (5.11), se deduce que las correlaciones entre el átomo óptimo $g_{k(i)}[n]$ en la iteración i-ésima y el resto de átomos $g_k[n] \in D$ también se pueden calcular mediante la FFT, en este caso conociendo la transformada de la ventana al cuadrado ($u[n] =$

$|w[n]|^2$). Por lo tanto, dicha transformada se tiene que pre-calcular y guardar en memoria para actualizar las correlaciones de forma directa.

El uso de un diccionario de exponenciales complejas permite [Verma99]: 1) calcular las correlaciones iniciales entre la señal y los átomos del diccionario mediante una FFT de longitud $2L$; 2) que las correlaciones cruzadas entre átomos del diccionario sólo requieran una memoria compleja de longitud $L+1$.

5.1.2. Extensión para el análisis de señales no estacionarias

Las señales no estacionarias se pueden analizar trama a trama, de forma que la señal se enventana en cada trama donde posee características estacionarias o, al menos, cambian poco. Una condición suficiente para asegurar la convergencia del modelo tonal mediante *matching pursuits* se expresa a continuación,

$$\sum_{l} w[n - lP] = 1 \tag{5.12}$$

donde $P \leq N$ representa el salto de señal entre tramas. Se pueden utilizar varios tipos de ventanas. La ventana más extendida es la ventana triangular [Verma99], pero esta elección conlleva un solapamiento que en la práctica provoca el incremento del número de tonos por muestra.

En esta tesis se propone el uso de ventanas que eviten por completo el solapamiento en (5.12). Para ello, se consideran ventanas rectangulares en el modelo tonal. Esta elección trae consigo la apariencia de un efecto de bloque (artefactos audibles) en las fronteras entre tramas, inconveniente que se solventa si en el receptor, a la hora de sintetizar la señal tonal, hay un pequeño solapamiento entre tramas adyacentes [Vera04b]. La forma de implementar esta idea es extendiendo en síntesis las tramas más allá de su duración en análisis, usando ventanas trapezoidales que suavicen la transición entre tramas. Con estas ventanas de síntesis, se consigue que los tonos que desaparecen de una trama a la siguiente no lo hagan de forma brusca, o los tonos que empiezan lo hagan de una forma suave. La mayor ventaja de este enfoque es que se evita el solapamiento en la etapa de análisis, siendo ésta una propiedad muy interesante para aplicaciones de codificación que usen el model tonal.

Una forma alternativa al uso de ventanas es el empleo de la interpolación de parámetros sinusoidales entre tramas de la señal de audio [George97]. Esta técnica hace uso de las trayectorias tonales que surgen en el modelo tonal, lo que significa que generalmente muchos tonos de una trama continúan generalmente en la siguiente, puesto que el modelo tonal trabaja con la parte armónica de la señal de audio. Para conseguir que el modelo pueda seguir las variaciones de la señal de entrada, se introduce la interpolación de parámetros en la ecuación (5.13),

$$x[n] \approx \hat{x}[n] = \sum_{k=1}^{K} A_k[n] \cdot cos\left(\omega_k[n] \cdot n + \phi_k[n]\right) \tag{5.13}$$

es decir, los valores $A_k[n]$, $\omega_k[n]$ y $\phi_k[n]$ se detectan en cada trama, se relacionan los tonos entre tramas adyacentes para formar trayectorias, y se modifican los parámetros entre una trama y otra siguiendo un procedimiento de interpolación que, generalmente, es lineal para la amplitud y polinómico para la frecuencia y fase [George97].

Cuadro 5.1: *Señales de audio utilizadas en el test del modelo tonal.*

SEÑAL	DESCRIPCIÓN
si01	Clavicordio
si03	Diapasón
sm01	Gaita
sm03	Punteos de guitarra

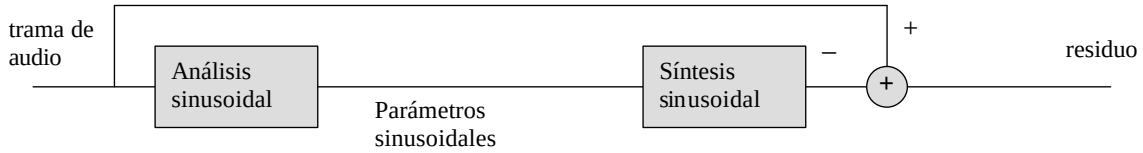

Figura 5.1: *Esquema experimental usado para comparar de forma objetiva diferentes métodos de implementación del modelo tonal.*

A continuación, se comparan ambos métodos de síntesis tonal, pero antes es necesario poner de manifiesto que, desde el punto de vista de la implementación, el tratamiento de ventanas es más rápido, puesto que es posible realizar la implementación en frecuencia mediante la FFT [Goodwin97]. Esta implementación se realiza sumando en frecuencia la transformada de cada tono extraído de la señal, realizando, una vez sumados todos los tonos, la transformada inversa y obteniendo la señal en el tiempo. Sin embargo, mediante la interpolación de parámetros, esto no es posible, puesto que al variar los parámetros (amplitud, frecuencia y fase) de cada tono con el tiempo n, es necesario realizar la síntesis de la señal en el tiempo mediante un oscilador. Como resultado, es necesario para implementar la interpolación de parámetros un banco de osciladores, donde hay un oscilador por cada frecuencia a sintetizar.

Para determinar el mejor método de análisis/síntesis para implementar el modelo sinusoidal, se han llevado a cabo varios experimentos con un conjunto de señales de prueba mono de calidad CD y alto contenido tonal. Las señales de audio elegidas son un subconjunto de las señales de prueba de MPEG-4 y se listan en la tabla 5.1.

La medida objetiva para comparar los diferentes métodos será la relación residuo a señal (RSR, *Residual to Signal energy Ratio*). Esta relación se calcula comparando la energía de la señal de entrada con el residuo obtenido tras aplicar el modelo tonal. Este residuo es el resultado de restar a la señal de entrada la señal salida del modelo tonal, es decir, la señal sintetizada tras extraer los parámetros con el modelo tonal. Un esquema de la obtención de esta señal residuo aparece en la figura 5.1.

Los diferentes métodos evaluados para las etapas de análisis y síntesis sinusoidal son:

A. Extracción tonal mediante búsqueda de picos espectrales en la DFT (*spectral peak picking*), y síntesis mediante interpolación de parámetros [Mcaulay86].

B. Extracción tonal mediante *matching pursuits*, y síntesis mediante interpolación de parámetros del modelo tonal.

C. Extracción tonal mediante *matching pursuits*, y uso de enventanado con ventana triangular tanto en análisis como en síntesis [George97][Verma99].

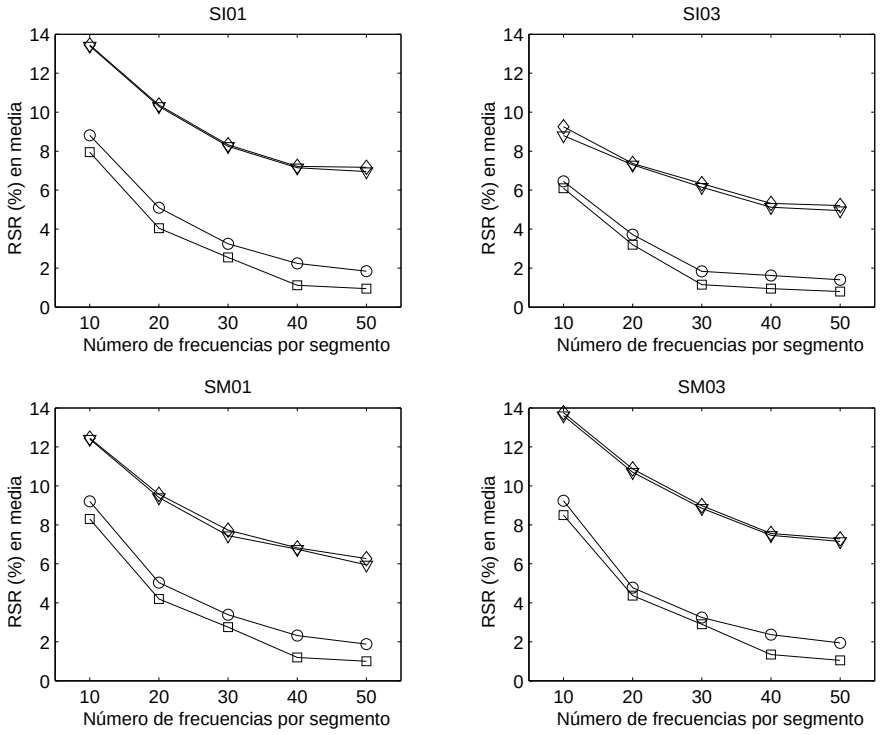

Figura 5.2: *Variación de la relación residuo a señal $RSR(\%)$ conforme aumenta el número de frecuencias extraídas para los cuatro métodos considerados: A (rombos), B (triángulos), C (círculos), D (cuadrados).*

D. Extracción tonal mediante *matching pursuits*, y uso de enventanado con ventana rectangular en análisis y trapezoidal (con un solapamiento del 10 %) en síntesis (método propuesto).

Los resultados que se muestran a continuación se han obtenido con un diccionario de exponenciales complejas con $L + 1 = 4097$ frecuencias y una longitud de trama de $N = 1024$ muestras.

En primer lugar, se comparan resultados con valores objetivos obtenidos por los métodos anteriormente citados. Así, en la figura 5.2 se muestran los valores medios de la relación residuo a señal, expresados en porcentaje $RSR(\%)$, conforme aumenta el número de frecuencias extraídas por el modelo sinusoidal.

Como se observa en la figura 5.2, los métodos C y D mejoran claramente los resultados de los métodos A y B. La diferencia entre ambos grupos está en el empleo de interpolación en síntesis (métodos A y B) frente al empleo de un esquema de enventanado (métodos C y D). Por lo tanto, una conclusión importante es que el uso de la interpolación de parámetros limita los resultados del modelo sinusoidal. En este sentido, la interpolación no es capaz de seguir las variaciones de la señal de entrada, lo que hace imposible reducir la relación residuo a señal. Al contrario, el enventanado es una técnica de reconstrucción perfecta, es decir, cuando el número de frecuencias tienda a infinito la relación residuo a señal tenderá a cero.

Una vez que se ha comprobado la ventaja de utilizar enventanado en términos objetivos, se pasa a comprobar qué ventanas es recomendable utilizar. En la bibliografía, es generalizado el uso de ventanas triangulares tanto en análisis como en síntesis cuando *matching pursuits* se utiliza

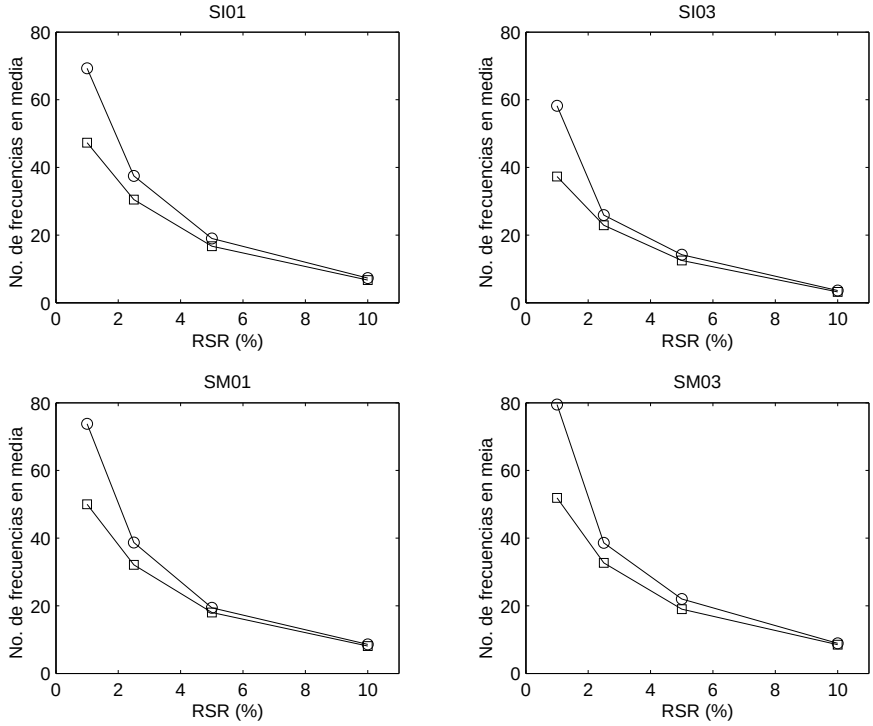

Figura 5.3: *Número de frecuencias necesarias para conseguir un valor fijo de relación residuo a señal* $RSR(\%)$ *para los métodos C (círculos) y D (cuadrados).*

como algoritmo de cálculo [Verma99] [Goodwin97]. Sin embargo, el inconveniente de este enfoque es el solapamiento producido por la ventana triangular, que es del 50 %. Este solapamiento incrementa el número de frecuencias por muestra y, como consecuencia, el régimen binario en aplicaciones de codificación. Como se ha dicho anteriormente, se propone el uso de ventanas rectangulares en análisis, que eviten el solapamiento, y trapezoidales en síntesis, que eviten el efecto de bloque entre segmentos. La comparación objetiva entre ambos métodos de ventanas se representa en la figura 5.3. Ahora, se dibuja el número de frecuencias que hay que extraer para lograr un valor dado de relación residuo a señal $RSR(\%)$ para los métodos C y D. Se puede apreciar como él método D propuesto consigue los mejores resultados.

La explicación de este resultado hay que buscarla en las ventanas de análisis. Parece lógico que la ventana rectangular en análisis consiga extraer más energía de la señal que la ventana triangular, lo que se traduce en una menor relación residuo a señal. Esto ocurre pese a que para el método D se cambie la ventana de síntesis a ventana trapezoidal. Claramente, este hecho debe de incrementar la relación RSR final, puesto que en las fronteras entre segmentos no se está sintetizando con la misma ventana con la que se analizó la señal. Finalmente, se puede afirmar que el empleo de la ventana rectangular en análisis compensa con creces el hecho de usar ventana trapezoidal en síntesis.

Para comprobar la calidad subjetiva del método propuesto, se propone cuantificar los parámetros tonales del bloque de análisis, según el esquema presentado en [Vera04b]. Para que la comparación entre métodos sea justa, se utilizará este mismo esquema de cuantificación para todos los métodos de modelado sinusoidal. El diagrama de bloques del sistema usado para comparar

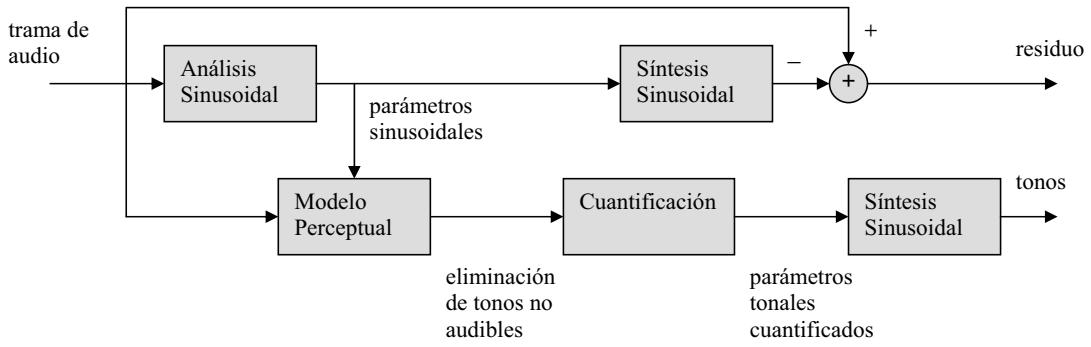

Figura 5.4: *Esquema experimental usado para comparar de forma subjetiva diferentes métodos de implementación del modelo tonal.*

la calidad subjetiva de los diferentes métodos aparece en la figura 5.4. Para medir la calidad subjetiva con este sistema, se comparará la señal original con la suma del residuo y la señal procedente del modelo tonal con parámetros cuantificados.

Los test de calidad subjetiva se han realizado entre 10 personas del departamento siguiendo la recomendación ITU-R BS.1116-1 [ITU-R97] para la evaluación de pequeñas desviaciones de calidad en señales de audio. Esta recomendación proporciona un resultado en una escala de 0 a 5 (escala MOS), tras escuchar el sujeto evaluador 3 veces las señales de test, 2 veces el original y 1 la señal evaluada, con referencia ciega, es decir, sin informarle de cuál es la señal original.

En la figura 5.5 se comparan los resultados subjetivos de los cuatro métodos considerados para el modelado sinusoidal, extrayendo 25 tonos por trama. Los valores subjetivos se han presentado en ΔMOS, que es la variación en calidad subjetiva entre el original y la suma de los tonos cuantificados y el residuo. Las mayores diferencias en calidad subjetiva aparecen con el uso de la interpolación (métodos A y B), puesto que este enfoque para la síntesis tonal no es capaz de seguir las variaciones rápidas de la señal de audio. Las variaciones entre métodos de enventanado son inapreciables.

Como conclusión, cabe decir que el uso de exponenciales complejas como elementos del diccionario del algoritmo *matching pursuits* permite una implementación eficiente basada en la FFT. Además, el uso del enventanado permite usar la FFT en la síntesis del modelo tonal. Se ha demostrado que el uso de ventana rectangular en análisis y trapezoidal en síntesis da lugar a los mejores resultados del modelo tonal y evita el solapamiento entre segmentos de análisis. Este esquema de funcionamiento proporciona una herramienta rápida y eficiente de implementar el modelo tonal, pero no tiene en cuenta en ningún momento principios psicoacústicos para extraer los tonos de la señal.

5.2. *Matching pursuits* con guiado perceptual

Una modificación deseable del método MP es la selección en cada iteración del átomo perceptualmente más importante. En el caso del modelo sinusoidal, esta estrategia se puede basar en

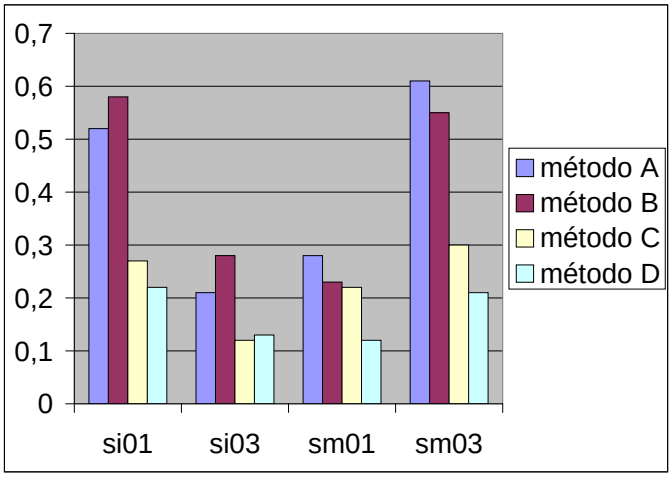

Figura 5.5: *Resultados subjetivos en ΔMOS comparando los métodos evaluados de modelado sinusoidal.*

principios psicoacústicos, ya que están bien estudiadas las propiedades del oído cuando la señal que lo excita es un tono estable [Zwicker90].

En todos los casos, el algoritmo *matching pursuits* escoge el átomo que tiene un peso asociado con mayor energía, por lo que se dice que está guiado por energía,

$$\max_{\mathbf{g}_k \in \mathbf{D}} ||\boldsymbol{\alpha}_k^i||^2 \tag{5.14}$$

ya sea para el método MP simple o extrayendo un subespacio conjugado, como ocurre con un diccionario de exponenciales complejas para implementar el modelo tonal.

Se define un algoritmo *matching pursuits* con guiado perceptual como aquel que escoge en cada iteración el átomo más importante psicoacústicamente, es decir,

$$\max_{\mathbf{g}_k \in \mathbf{D}} ||\boldsymbol{\alpha}_k^i||_{perceptual} \tag{5.15}$$

Por lo tanto, es necesario definir una medida perceptual, a partir del valor del peso de cada átomo en el método MP convencional (guiado por energía). En este sentido, han aparecido en la bibliografía varias propuestas, cada una de ellas con problemas asociados, que se comentarán a continuación.

5.2.1. *Weighted Matching Pursuits*

La primera propuesta de modificar el método MP para incluir información psicoacústica aparece en [Verma99b], y se conoce con el nombre de *Weighted Matching Pursuits* (WMP). En este caso, la medida perceptual se reduce simplemente a modificar cada peso α_k^i de la forma,

$$||\boldsymbol{\alpha}_k^i||_{WMP} = \frac{|\boldsymbol{\alpha}_k^i|^2 |G[k]|^2}{T[k]} \tag{5.16}$$

donde $G[k]$ es transformada discreta de Fourier del átomo $\mathbf{g}_k$ y $T[k]$ el umbral de enmascaramiento en la frecuencia k. Esta medida perceptual simplemente modifica la energía por la relación señal a máscara en la frecuencia de la exponencial compleja k. Uno de los inconvenientes

de este enfoque es que sólo está pensado para su uso con exponenciales complejas, ya que sólo evalúa la relación señal a máscara en el valor de la frecuencia k. Como ventaja, está la rapidez en el cálculo de la medida perceptual.

5.2.2. *Psychoacoustic-Adaptive Matching Pursuits*

Una mejora con respecto a WMP se presenta en [Heusdens02] y es conocida como *Psychoacoustic-Adaptive Matching Pursuits* (PAMP). La mejora consiste en sustituir la relación señal a máscara a la frecuencia del tono por el área comprendida entre el tono y la máscara, puesto que el oído es un elemento que evalúa la señal en todas las frecuencias, no sólo en la frecuencia del tono de entrada. De esta forma, la medida perceptual PAMP queda,

$$||\boldsymbol{\alpha}_k^i||_{PAMP} = \int_0^1 \frac{|\boldsymbol{\alpha}_k^i|^2 |G_k(f)|^2}{T(f)]} df \tag{5.17}$$

La ventaja de PAMP frente a WMP se pone de manifiesto cuando la trama de análisis de la señal es de longitud finita [Heusdens02]. En este caso, la transformada de cada exponencial compleja no es una simple delta en frecuencia, sino la transformada de la ventana utilizada. Por tanto, la importancia perceptual hay que calcularla, no solo a la frecuencia nominal k del tono, sino en toda la frecuencia. Esta ventaja se pone de manifiesto en la figura 5.6, donde se representa la extracción tonal de dos tonos de frecuencias 1 kHz y 1,1 kHz. En la primera gráfica de la figura 5.6 se observa el espectro del tono de 1 kHz y la máscara que genera el tono de 1,1 kHz, ya extraído. En la siguiente gráfica se representa la medida perceptual, según WMP, en la primera iteración (cuando ya ha sido extraída la frecuencia de 1,1 kHz), es decir, $||\alpha_k^2||_{WMP}$. Finalmente, en la última gráfica se presenta la medida perceptual para PAMP, $||\alpha_k^2||_{PAMP}$. Se puede observar cómo para WMP el máximo de la medida perceptual no se produce en 1 kHz, debido a que la medida perceptual es función de la diferencia entre el espectro de potencia del tono de 1 kHz y la máscara generada por el tono de 1,1 kHz. En PAMP, en cambio, la medida perceptual es el área entre un tono centrado en la frecuencia k y la máscara actual. Como se observa en la figura, la definición PAMP de la medida perceptual permite seleccionar como tono más importante perceptualmente al tono de 1 kHz, aunque se haya extraído anteriormente un tono cercano en frecuencia.

Aunque este resultado demuestra el correcto funcionamiento de la selección PAMP de los tonos, esta manera de calcular la importancia perceptual de los átomos también tiene sus problemas asociados:

- Por un lado, en el diseño de la medida PAMP no se tiene en cuenta el tratamiento que realiza el oído cuando tiene que procesar una señal sonora. Un modelo simplificado del funcionamiento del oído aparece en la figura 5.7. En esta figura, basada en la modelización del procesado del oído como sistema lineal realizada en [Par02], se asume que la respuesta del oído externo y medio es simplemente un filtro lineal que tiene una forma similar al inverso del umbral de silencio. El oído interno, y concretamente la membrana basilar, realiza un filtrado paso banda con un banco de filtros en bandas críticas, es decir, un banco de filtros donde el ancho de banda de cada filtro crece con la frecuencia. Posteriormente, la excitación de cada banda se compone para formar la sensación auditiva. Como consecuencia de este tratamiento, las señales de baja frecuencia producen una sensación auditiva más

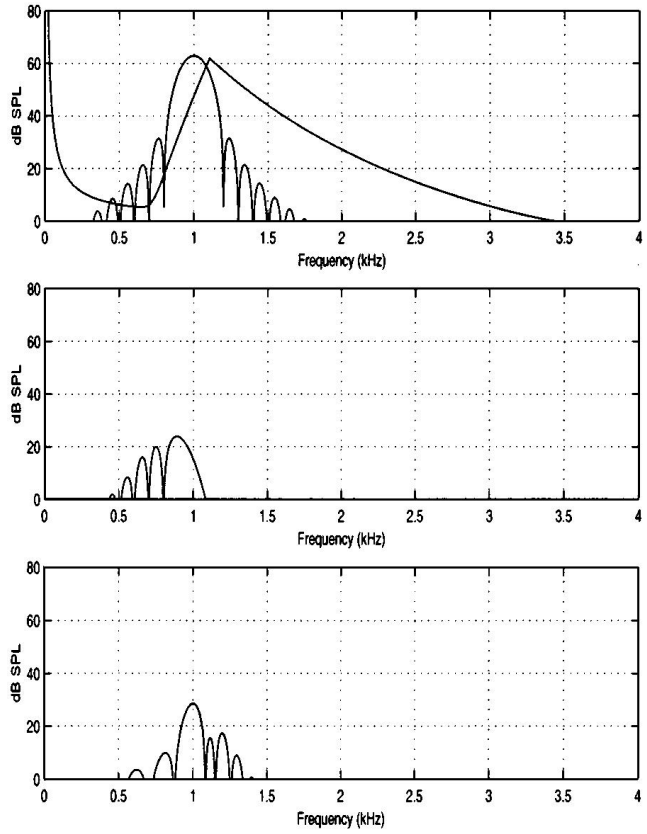

Figura 5.6: *Ejemplo de funcionamiento de las medidas perceptuales WMP y PAMP para el caso de dos tonos de* $1kHz$ *y* $1,1kHz$ *[Heusdens02]. Se presenta primero el espectro de potencia de un tono de* $1kHz$ *junto con la máscara generada por el tono de* $1,1kHz$*, después la medida perceptual WMP tras extraer el tono de* $1,1kHz$ *y, por último, la medida perceptual PAMP.*

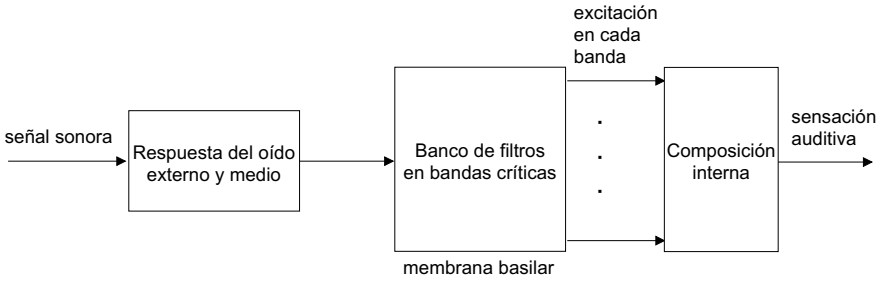

Figura 5.7: *Modelo del oído como sistema lineal.*

fuerte, puesto que excitan un mayor número de bandas críticas que las señales de alta frecuencia. Este comportamiento logarítmico en la frecuencia es obviado por la medida perceptual PAMP.

- Por otro lado, desde el punto de vista de la implementación práctica, la medida PAMP tiene un alto coste computacional. Esto se debe a que para cada átomo o frecuencia k es necesario realizar un sumatorio de todos los valores de la relación en frecuencia entre señal y la máscara actual, siendo este proceso mucho más costoso que la simple operación por frecuencia propuesta en la selección WMP. Además, la obtención de la máscara en frecuencia tiene un coste computacional adicional, puesto que normalmente los modelos psicoacústicos trabajan con la señal estimada en bandas críticas para calcular la máscara [MPEG92]. Por tanto, es necesario trabajar con la señal en bandas críticas para calcular el umbral de enmascaramiento en frecuencia logarítmica y, posteriormente, trasladar los resultados de la máscara a frecuencia lineal.

5.2.3. *Perceptual Matching Pursuits*

A la vista de estos problemas, una forma de mejorar los resultados de la medida perceptual PAMP es realizar la integración en banda de Bark en lugar de en frecuencia [Vera06b]. De esta forma, se consigue tener en cuenta el procesado que se produce en el oído interno y, a la vez, reducir la complejidad computacional. La medida perceptual propuesta se ha llamado PMP *Perceptual Matching Pursuits*, y se calcula mediante la ecuación (5.18),

$$||\boldsymbol{\alpha}_k^i||_{PMP} = \int_0^B \frac{|\boldsymbol{\alpha}_k^i|^2 |G_k(b)|^2}{T(b)} db \tag{5.18}$$

donde b denota banda de Bark, B es la máxima banda de Bark de la señal de entrada (este valor depende de la frecuencia de muestreo de la señal), $|G_k(b)|^2$ es el espectro de potencia del átomo $\mathbf{g}_k$ en banda de Bark y $T(b)$ es la máscara en banda de bark.

Para verificar el correcto funcionamiento de la medida perceptual propuesta PMP, se realiza primero la prueba de los dos tonos para comprobar que realiza la selección correctamente y, también, para analizar las diferencias de resultados entre ambas estrategias. En la figura 5.8 se representa en (a) los pesos $||\boldsymbol{\alpha}_k^2||^2$, tras extraer el tono de 1,1 kHz. Como es lógico, el valor de el peso para la frecuencia de 1,1 kHz ha de ser cero para cumplir la ortogonalidad que se produce en MP entre el átomo extraído y el residuo de la siguiente iteración. Se puede observar cómo

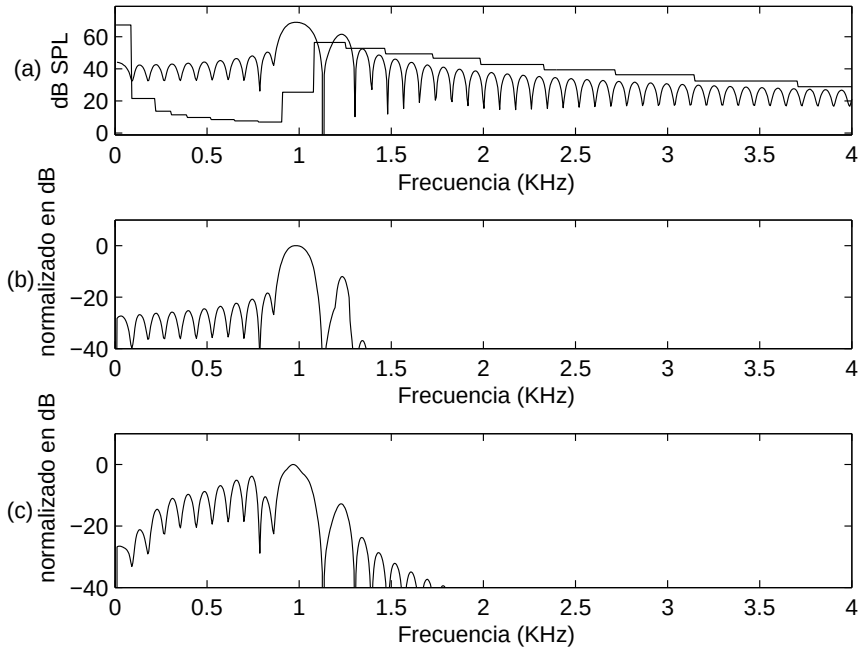

Figura 5.8: *Ejemplo de funcionamiento de las medidas perceptuales PAMP y PMP para el caso de dos tonos de* $1kHz$ *y* $1,1kHz$*. (a) Peso de los átomos y máscara para la iteración* $i = 2$*, (b) medida perceptual PAMP para la iteración* $i = 2$ *y (c) medida perceptual PMP para la iteración* $i = 2$*.*

el método MP extrae toda la correlación de la señal con la frecuencia de 1,1 kHz, que para el ejemplo corresponde al tono inicial de 1,1 kHz, más el valor que produce la frecuencia de 1 kHz en 1,1 kHz debido a la transformada de la ventana. En (b) se dibujan la medida perceptual que proporciona PAMP integrando en frecuencia, $||\alpha_k^2||_{PAMP}$, tras extraer el primer tono, mientras que en (c) se dibuja la medida perceptual de PMP integrando en banda de bark, $||\alpha_k^2||_{PMP}$. En general, para ambas medidas se extraerá correctamente el tono de 1 kHz. Se observa como en la medida PMP en banda de bark las bajas frecuencias tienen un mayor peso perceptual que las altas frecuencias en relación a la medida en PAMP.

La medida PMP, al estar definida en banda de Bark, simula mejor el comportamiento del oído, lo que debe redundar en un mejor funcionamiento, sobre todo, en relación al ruido, como se verá más adelante. Pero, además, esta definición disminuye considerablemente el número de operaciones para obtener la medida perceptual. Así, para la medida PAMP, por cada exponencial compleja es necesario, según la ecuación (5.17), la suma de todos los valores en frecuencia resultantes de multiplicar el peso por la transformada al cuadrado y dividirlo entre la máscara. Es importante notar que la frecuencia debe ser muestreada en una implementación práctica. Si se realiza la inversa de la máscara en frecuencia a priori, para cada exponencial compleja hay que realizar dos multiplicaciones por cada muestra en frecuencia y sumar todos estos valores. El problema viene determinado por el muestreo que hay que realizar en frecuencia para implementar la ecuación (5.17). Como la resolución en frecuencia del oído es logarítmica, para tener una resolución aceptable haciendo un muestreo uniforme en frecuencia, es necesario coger una separación en frecuencia que da lugar a un gran número de muestras. Así, el valor de *Just Noticeable*

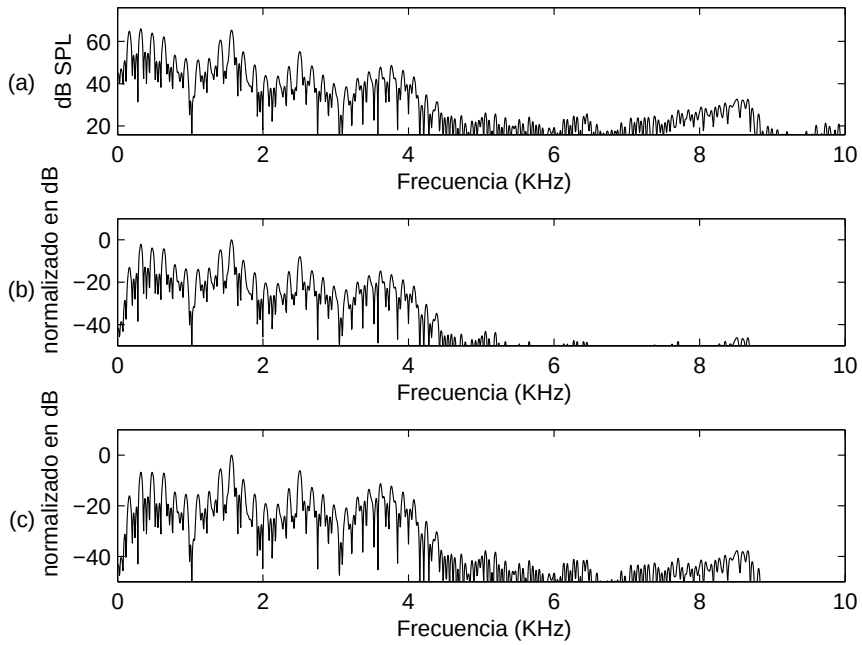

Figura 5.9: *Ejemplo de funcionamiento de las medidas perceptuales PAMP y PMP para el caso una señal vocal sonora. (a) Espectro de energía de la señal de entrada, (b) medida perceptual PMP para la iteración inicial y (c) medida perceptual PAMP para la iteración inicial.*

Difference para distinguir dos frecuencias es de 0,2 Bark para 10 ms y 0,01 Bark para 500 ms [Zwicker90]. En la práctica estos valores son de 4096 muestras para un tamaño de segmento de 45 ms y una frecuencia de muestreo de 44.100 kHz. Sin embargo, de acuerdo con la selección PMP, el muestreo según la ecuación (5.18) se realiza en banda de Bark. Como la resolución se adapta ahora en la misma forma en que *escucha* el oído humano, el número de muestras se reduce considerablemente. Por ejemplo, para el modelo de enmascaramiento de MPEG [MPEG92], la resolución empleada al realizar el muestreo es de 3 muestras por banda de Bark, lo que resulta en menos de 60 muestras para una frecuencia de muestreo de 44.100 kHz. Como consecuencia de todo esto, la complejidad de calcular la medida PMP es, en la práctica, bastante menor que para PAMP. Básicamente, esto es debido al alto número de muestras que debe utilizar PAMP para implementar un muestreo lineal de la señal en frecuencia, que no está adaptado al comportamiento del oído humano.

A continuación, se comentan las diferencias entre los resultados obtenidos por ambos métodos, sin entrar en la complejidad que conllevan, sino para analizar el comportamiento cuando la señal de entrada es una señal vocal sonora, así como para comprobar la capacidad de discriminación frente al ruido de cada forma de calcular la medida perceptual. En primer lugar, se van a presentar los resultados de ambos métodos cuando la señal a analizar es una señal vocal sonora masculina. En la figura 5.9, se representa en (a) el espectro de potencia de la señal vocal, en (b) la medida perceptual PMP (en banda de Bark) en la iteración inicial, $||\alpha_k^1||_{PMP}$, y en (c) la medida perceptual PAMP (en frecuencia) en la iteración inicial, $||\alpha_k^1||_{PAMP}$. La única diferencia apreciable es que para PMP la importancia perceptual de las altas frecuencias se reduce, puesto que su energía se concentra en unas pocas bandas de Bark.

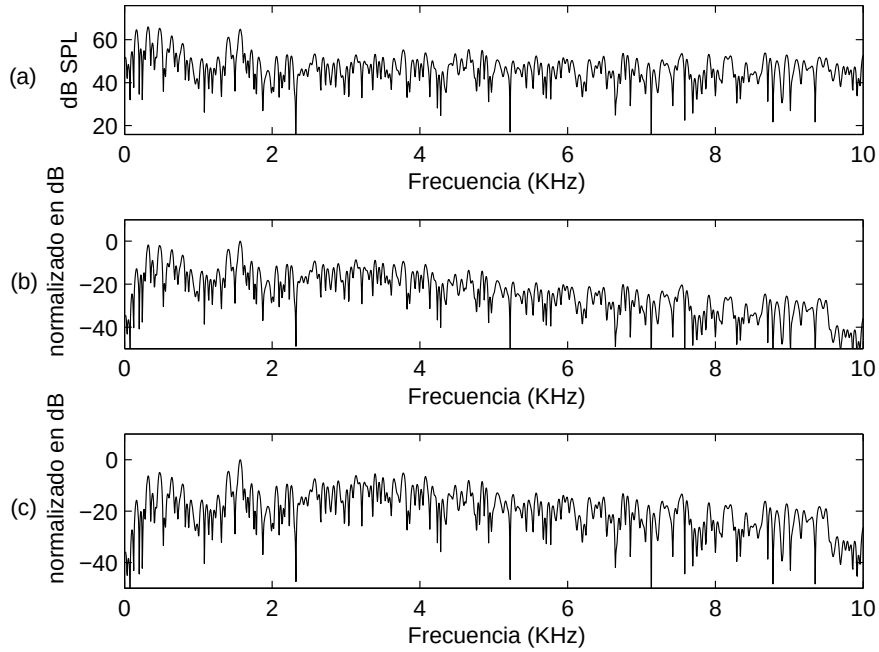

Figura 5.10: *Ejemplo de funcionamiento de las medidas perceptuales PAMP y PMP para el caso una señal vocal sonora más ruido blanco. (a) Espectro de energía de la señal de entrada, (b) medida perceptual PMP para la iteración inicial y (c) medida perceptual PAMP para la iteración inicial.*

El siguiente paso es comprobar la discriminación frente al ruido de ambas medidas perceptuales. Para ello, se suma a la señal anterior la misma energía de ruido blanco (relación señal a ruido de 0 dB) y se comprueban los valores de las medidas perceptuales correspondientes en la figura 5.10. En este caso, la cantidad de ruido hace indistinguibles los tonos del ruido en las frecuencias de 2 a 4 kHz para PMP y PAMP, aunque para PMP los valores perceptuales en esta banda son menores que para PAMP, en relación a los tonos que se aprecian de 0 a 2 kHz. Además, las altas frecuencias tienen valores perceptuales menores para PMP. En particular, se aprecia una mayor caída de la medida perceptual con la frecuencia.

En las diferentes estrategias de cálculo de una medida perceptual aparece el umbral de enmascaramiento como parte integrante de la definición. Sin embargo, de momento, no se ha comentado nada acerca de cómo calcular este umbral. Como primera aproximación, se podría pensar en utilizar el clásico umbral de enmascaramiento de ruido que utilizan los codificadores por transformada. Es umbral informa de cuánto ruido de cuantificación es posible inyectar en cada frecuencia o banda del codificador. Sin embargo, para la aplicación que nos ocupa este umbral no es el idóneo, puesto que se trata de determinar los tonos perceptualmente más importantes. La solución utilizada en la bibliografía para calcular las distintas medidas perceptuales consiste en utilizar el umbral de silencio como punto de partida, es decir, el umbral de silencio es el umbral que se utiliza en la iteración inicial del método *matching pursuits* [Heusdens02]. El umbral para las siguientes iteraciones se actualiza a partir de la máscara que genera el tono extraído en cada iteración, $\boldsymbol{\alpha}_{k(i)}\mathbf{g}_{k(i)}$. Así pues, esta definición del umbral de enmascaramiento para medidas perceptuales queda de la forma,

$$T^i(b) = \begin{cases} T_{quiet}(b) & i = 0 \\ T^{i-1}(b) + T_{\boldsymbol{\alpha}_{k(i)}\mathbf{g}_{k(i)}} & i > 0 \end{cases} \tag{5.19}$$

donde $T_{quiet}(b)$ es el umbral de silencio, $T_{\boldsymbol{\alpha}_{k(i)}\mathbf{g}_{k(i)}}$ es la máscara debida al tono extraído en la iteración i y b denota banda de Bark. Es preciso tener en cuenta que ahora el umbral de enmascaramiento se escribe $T^i(b)$, puesto que depende de la iteración en que se encuentre el método *matching pursuits*. Se ha escrito el umbral en banda de bark, puesto que es la forma general de calcularlo, aunque para WMP y PAMP sea preciso posteriormente cambiarlo a frecuencia lineal. Se ha utilizado la suma para la composición de umbrales, por ser la forma más fácil de realizar la notación, aunque existen otras opciones más conservadoras de calcular esta composición [Zwicker90]. Este cálculo de la máscara es el que se ha utilizado en todas las figuras presentadas hasta ahora en este capítulo. Por eso, en la figura 5.6, el umbral, una vez extraído el primer tono, incluye el triángulo centrado en la frecuencia 1,1 kHz, que representa la máscara de este tono. El umbral de silencio es mucho mayor para esta figura en las frecuencias cercanas a cero. Para la figura 5.6, el umbral se calcula directamente en frecuencia, mientras que para la figura 5.8 ya se calcula en banda de Bark, por lo que al dibujar la máscara en un eje de frecuencias, queda una señal escalonada con tamaños de escalón mayores para alta frecuencia. Con esta definición del umbral la ecuación que define la medida PMP se puede escribir como,

$$||\boldsymbol{\alpha}_k^i||_{PMP} = \int_0^B \frac{|\boldsymbol{\alpha}_k^i|^2 |G_k(b)|^2}{T^{i-1}(b)} db \tag{5.20}$$

Análogamente, la misma modificación para el umbral habría que realizarla en las ecuaciones (5.16) y (5.17).

La inicialización de la máscara al umbral de silencio no proporciona, sin embargo, una característica deseable para la extracción tonal, que es una parada psicoacústica. Esto quiere decir que el método MP debe detenerse cuando se hayan extraído todos los tonos audibles. Para saber cuándo parar el algoritmo, según [Zwicker90], un tono puede ser enmascarado por el umbral de silencio, otro tono con mayor intensidad sonora o por ruido. Si se inicializa al umbral de silencio, no se extraerán los tonos enmascarados por este umbral, y con el paso de las iteraciones los tonos extraídos enmascararán a otros tonos de la señal, pero los tonos enmascarados por ruido se extraerán como tonos audibles. La solución a este inconveniente consiste en inicializar la máscara al umbral de silencio más el umbral de ruido sobre tonos, NMT (*Noise-is-Masking-Tone*). En general, en la mayoría de los modelos perceptuales, para discriminar entre la parte tonal (o predecible) de la señal y la parte ruidosa (o impredecible) se utiliza un índice de tonalidad [Brandenburg90]. Este índice de tonalidad se usa por un lado para generar la máscara producida por los tonos, y por otro, para obtener la máscara producida por el ruido. Para calcular correctamente una máscara de ruido sobre tonos, NMT, se aplica el índice de tonalidad del modelo de enmascaramiento, para posteriormente generar sólo la máscara de ruido sobre tonos, no incluyendo la máscara de tonos sobre tonos. Por lo tanto, la inicialización de la máscara a utilizar propuesta queda expresada de la forma,

$$T^i(b) = \begin{cases} T_{quiet}(b) + T_{NMT}(b) & i = 0 \\ T^{i-1}(b) + T_{\boldsymbol{\alpha}_{k(i)}\mathbf{g}_{k(i)}} & i > 0 \end{cases} \tag{5.21}$$

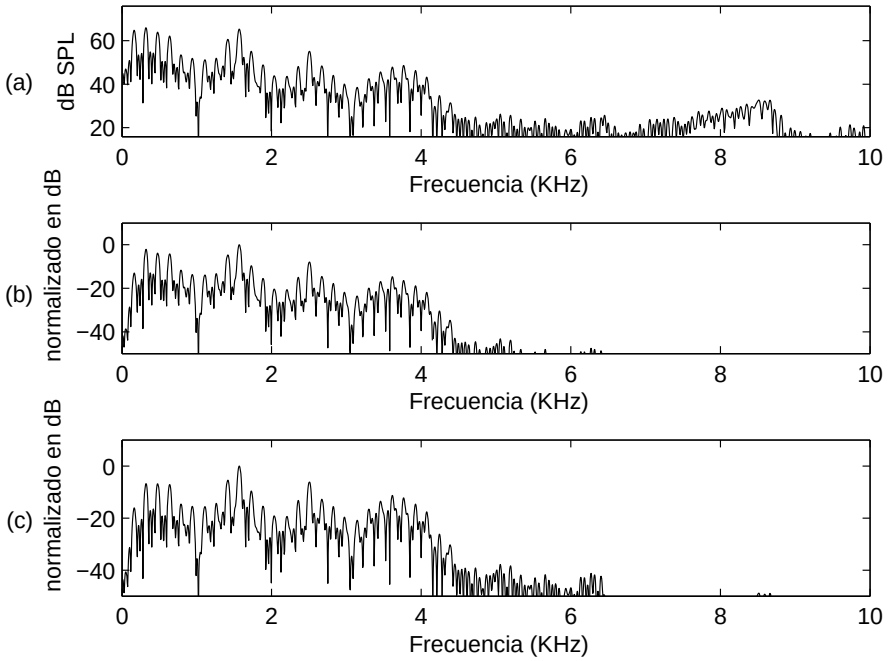

Figura 5.11: *Ejemplo de funcionamiento de las medidas perceptuales PAMP y PMP para el caso una señal vocal sonora con máscara inicial que incluye el umbral NMT. (a) Espectro de energía de la señal de entrada, (b) medida perceptual PMP para la iteración inicial y (c) medida perceptual PAMP para la iteración inicial.*

En primer lugar, se incluyen de nuevo las figuras correspondientes a las medidas perceptuales PMP y PAMP para voz sonora y para voz sonora más ruido, con el objetivo de comprobar las variaciones que se producen en las medidas perceptuales al incluir el umbral NMT. En todas las figuras se calcula el umbral NMT, realizando las modificaciones explicadas anteriormente al modelo de enmascaramiento propuesto por MPEG [MPEG92]. En la figura 5.11 se aprecia el resultado de aplicar la máscara NMT en la inicialización. La única diferencia con respecto a los resultados de la figura 5.9 es la disminución de la importancia perceptual de los tonos a partir de 7 kHz. Esta disminución se debe a que en esta zona de frecuencia el valor del umbral NMT tiene relativa importancia respecto al umbral de silencio. Esto ocurre porque el índice de tonalidad ha estimado una señal ruidosa en estas frecuencias.

Estos resultados se ven con mayor énfasis cuando se añade ruido blanco a la señal. Así, en la figura 5.12, donde se incluye la máscara NMT con respecto a lo presentado en la figura 5.10, la importancia perceptual baja en ciertas bandas que es donde el índice de tonalidad estima que predomina el ruido sobre los tonos. Sin embargo, el índice de tonalidad no estima ruido en bandas de alta frecuencia donde sí lo hay. En general, puede afirmarse que sí permite una mejor discriminación de la señal con respecto al ruido para las medidas perceptuales PMP y PAMP. Sería interesante, a la vista de los resultados, estimar la máscara inicial mediante modelos de enmascaramiento que no utilizan un índice de tonalidad, como para el modelo propuesto en [Par02]. En un modelo de este tipo, debería obtener una mejor discriminación entre tonos y ruido.

La ventaja fundamental de incluir en la máscara el umbral NMT es la posibilidad de imple-

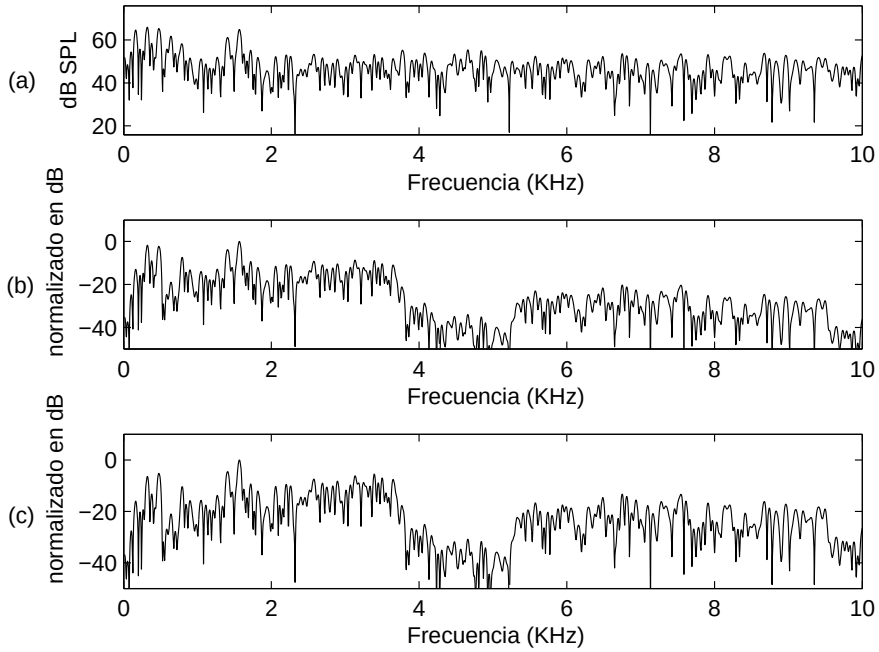

Figura 5.12: *Ejemplo de funcionamiento de las medidas perceptuales PAMP y PMP para el caso una señal vocal sonora más ruido blanco con máscara inicial que incluye el umbral NMT. (a) Espectro de energía de la señal de entrada, (b) medida perceptual PMP para la iteración inicial y (c) medida perceptual PAMP para la iteración inicial.*

mentar una parada psicoacústica. Aunque en algunas aplicaciones no es posible tener un régimen binario que permita extraer todos los tonos perceptualmente significativos, esta posibilidad es interesante si se desea conocer el número máximo de tonos a extraer para conseguir la máxima calidad posible en un codificador paramétrico. La parada perceptual del método *matching pursuits* guiado perceptualmente se debe producir cuando se extraen todos los tonos que están por encima de la máscara. En general, esta parada se produce en la iteración en la que todos los tonos están por debajo de la máscara, es decir, cuando se cumple,

$$|\boldsymbol{\alpha}_k^i|^2 \cdot |G_k(b)|^2 \leq T^{i-1}(b), \quad b = [0, B] \; \forall \mathbf{g}_k \in D \tag{5.22}$$

Cuando esta condición es cierta, los tonos que quedan en el residuo no son audibles y se detiene el método MP.

Para comprobar la viabilidad de la parada perceptual, se probará con una señal vocal sonora para las medidas perceptuales PMP y PAMP. Para PMP, en la figura 5.13, se dibuja en (a) el espectro de energía de la señal en la primera columna y la medida perceptual en banda de bark para la primera iteración $||\alpha_k^1||_{PMP}$ en la segunda columna. En (b) se presentan los mismos resultados correspondientes a la iteración $i = 2$, donde se ha añadido además marcado con un círculo el tono extraído. Es curioso observar las variaciones que se producen tanto en energía como en medida perceptual. En energía, al utilizar *matching pursuits*, desaparece del residuo todo el lóbulo del tono extraído, siendo sustituida esta información por los parámetros de amplitud, frecuencia y fase del tono. En medida perceptual los cambios son mucho mayores. Al sumarse a la máscara actual la máscara debida al tono extraído, todos los valores en frecuencia inmediatamente

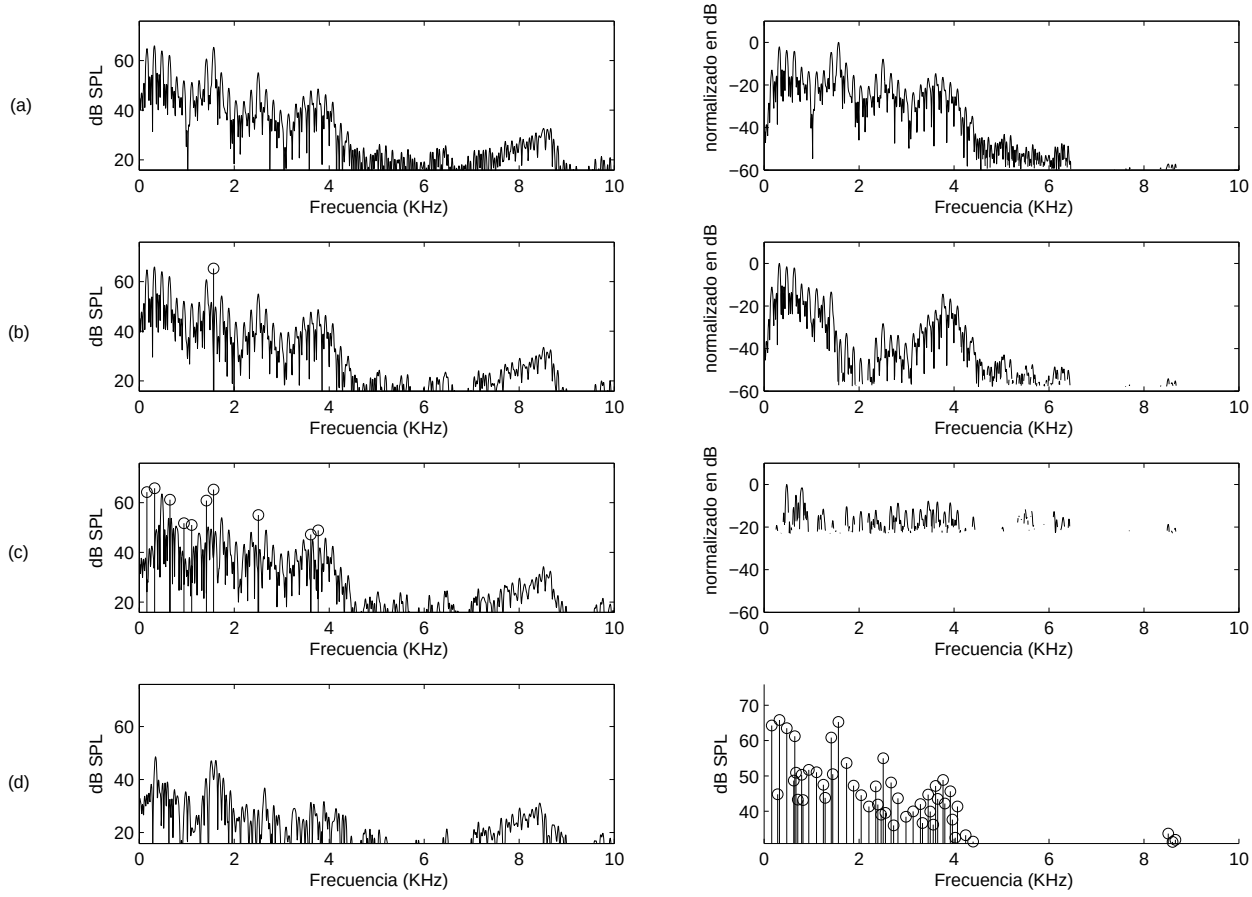

Figura 5.13: *Ejemplo de funcionamiento de la parada perceptual con la medida PMP para el caso una señal vocal sonora. (a) Espectro de energía de la señal de entrada (primera columna) y medida perceptual inicial (segunda columna), (b) espectro de energía del residuo (se incluyen los parámetros del tono extraído usando una recta terminada en círculo) y medida perceptual PMP para la iteración* $i = 2$*, (c) iteración* $i = 11$ *y (d) residuo final y tonos extraídos aplicando la parada perceptual .*

posteriores sufren una importante bajada en medida perceptual. Esto se debe a que la máscara de un tono tiene forma triangular en frecuencia, pero con una caída mucho más suave para las frecuencias mayores que la del tono extraído. En la gráfica no se han dibujado aquellos tonos que están por debajo de la máscara, según la condición de la ecuación (5.22). Los tonos que no tienen dibujada medida perceptual se corresponden con esta situación. En (c) se presentan los mismos resultados para la iteración $i = 11$. Se puede apreciar cómo se reduce el número de tonos por encima de la máscara conforme se extraen tonos, debido a las máscaras generadas por los tonos ya extraídos. Se observa, además, cómo los tonos de alta frecuencia (entre 8 y 9 kHz en la gráfica) no resultan enmascarados por los tonos de baja frecuencia extraídos. Finalmente, en (d) se representa el residuo final de la iteración $i = 47$ (en la primera columna), que es cuando todos los tonos están por debajo de la máscara; en la segunda columna se representan todos los tonos extraídos, donde se aprecia que se han extraído también los tonos de alta frecuencia.

La parada perceptual también funciona para el caso de la medida perceptual en frecuencia PAMP. Los resultados con este enfoque se han representado en la figura 5.14. Aunque se extrae

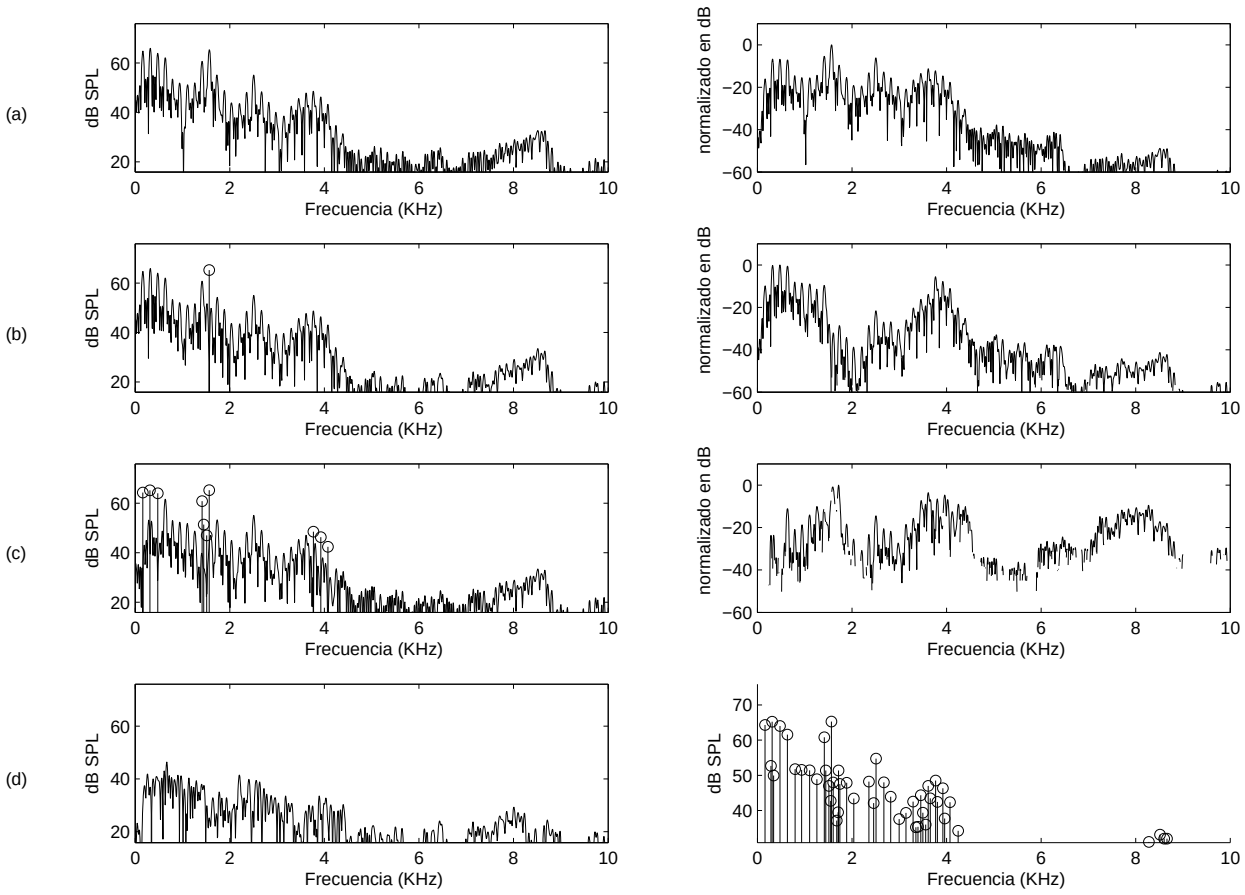

Figura 5.14: *Ejemplo de funcionamiento de la parada perceptual con la medida PAMP para el caso una señal vocal sonora. (a) Espectro de energía de la señal de entrada (primera columna) y medida perceptual inicial (segunda columna), (b) espectro de energía del residuo (se incluyen los parámetros del tono extraído usando una recta terminada en círculo) y medida perceptual PAMP para la iteración $i = 2$, (c) iteración $i = 11$ y (d) residuo final y tonos extraídos aplicando la parada perceptual .*

primero el mismo tono, es significativo que los caminos de ambas medidas son diferentes, no extrayéndose los tonos en el mismo orden. Incluso en la iteración final, que en este caso se produce en la iteración $i = 49$, se observa que los tonos extraídos no han sido exactamente los mismos, si bien esto ocurre para tonos de baja energía muy cercanos en frecuencia a otros tonos de mayor energía. Una diferencia importante está en las altas frecuencias, se aprecia como para PAMP se extrae algún tono más de alta frecuencia.

Una vez analizado el comportamiento de las medidas perceptuales en las sucesivas iteraciones, es el momento de comprobar cómo afecta el ruido blanco a PMP y PAMP. Con este fin, se incluye la figura 5.15, donde se dibuja el residuo y la medida perceptual en frecuencia para la iteración (a) inicial, (b) $i = 2$, y (c) $i = 5$. Se observa cómo tras extraer el primer tono la importancia perceptual de las frecuencias en torno a 2 kHz cae bruscamente. Para las frecuencias de 5 a 6 kHz la medida perceptual ya tenía un valor bajo debido al índice de tonalidad. Sin embargo, la zona entre 6 y 9 kHz tiene una importancia perceptual relativamente alta, por lo que en la quinta iteración ya se ha extraído un supuesto tono de alta frecuencia debido al ruido que se ha

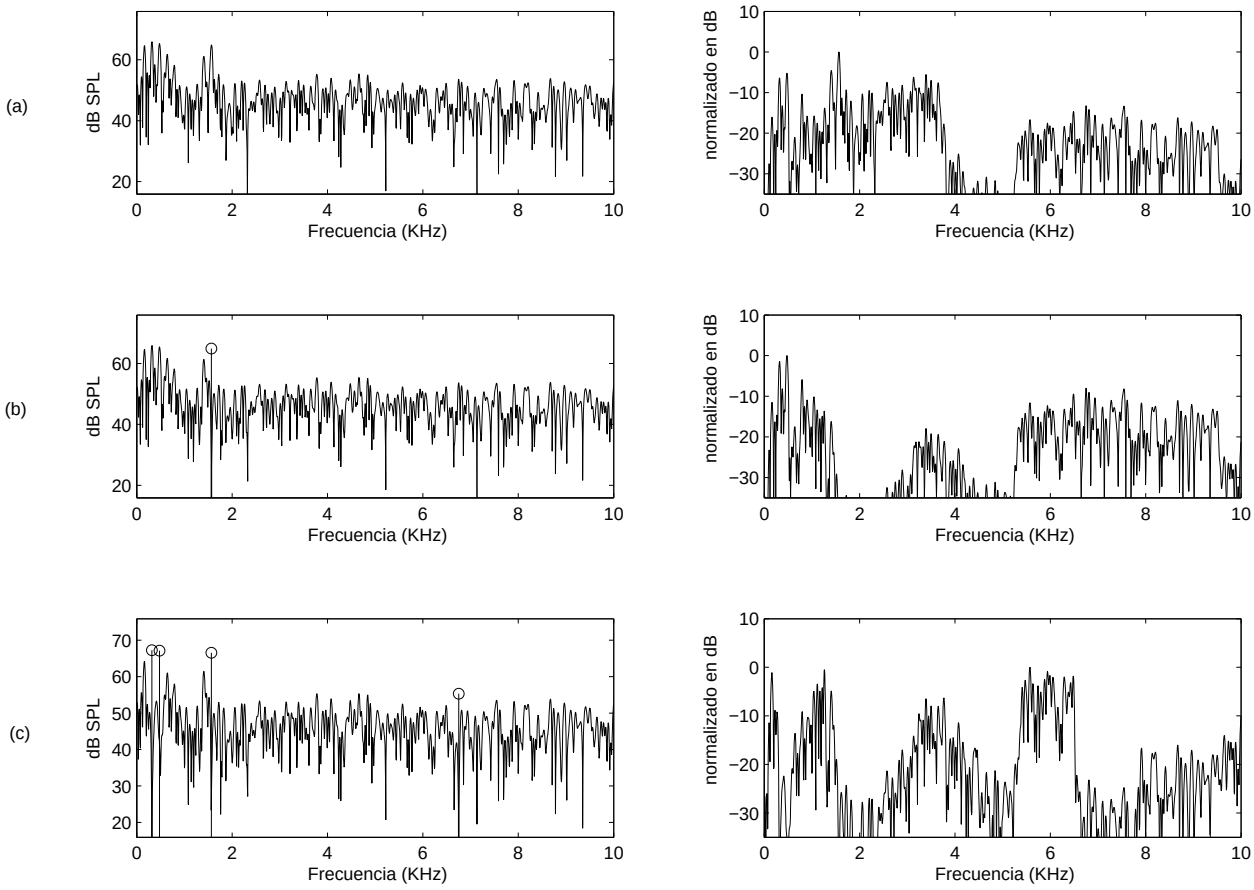

Figura 5.15: *Ejemplo de funcionamiento de la medida PAMP para el caso una señal vocal sonora más ruido. (a) Espectro de energía de la señal de entrada (primera columna) y medida perceptual inicial (segunda columna), (b) espectro de energía del residuo (se indican los parámetros del tono extraído usando una recta terminada en círculo) y medida perceptual PMP para la iteración $i = 2$, (c) iteración $i = 5$.*

sumado a la señal. Este tono se extrae antes que varios de los tonos de 0 a 2 kHz, que aún se distinguen en la señal con ruido.

Para tratar el caso de PMP en una señal vocal con ruido, se ha incluido la figura 5.16. En este caso, se ha dibujado la misma señal en las mismas iteraciones, siendo, el resultado diferente. Ya en la medida perceptual inicial, los valores de importancia perceptual de 6 a 9 kHz son significativamente menores que en el caso anterior. El cambio de la medida perceptual de frecuencia (PAMP) a banda de Bark (PMP) permite la extracción de tonos de baja frecuencia en las primeras cuatro iteraciones. Incluso después, como se aprecia en la figura 5.16 (c), se va a extraer otro tono de baja frecuencia, lo que se puede afirmar al localizar el máximo de la medida perceptual. En las siguientes iteraciones, será inevitable la extracción de algún tono en la zona de ruido, aunque esto se producirá más tarde que en el caso de PAMP, y cuando ya se han extraído casi todos los tonos de baja frecuencia. Como conclusión, cabe decir que, en general, el cálculo de la medida perceptual en banda de bark (PMP) proporciona una medida más adecuada para discriminar tonos de ruido, puesto que en las señales de audio los tonos más importantes se encuentran en la media y baja frecuencia.

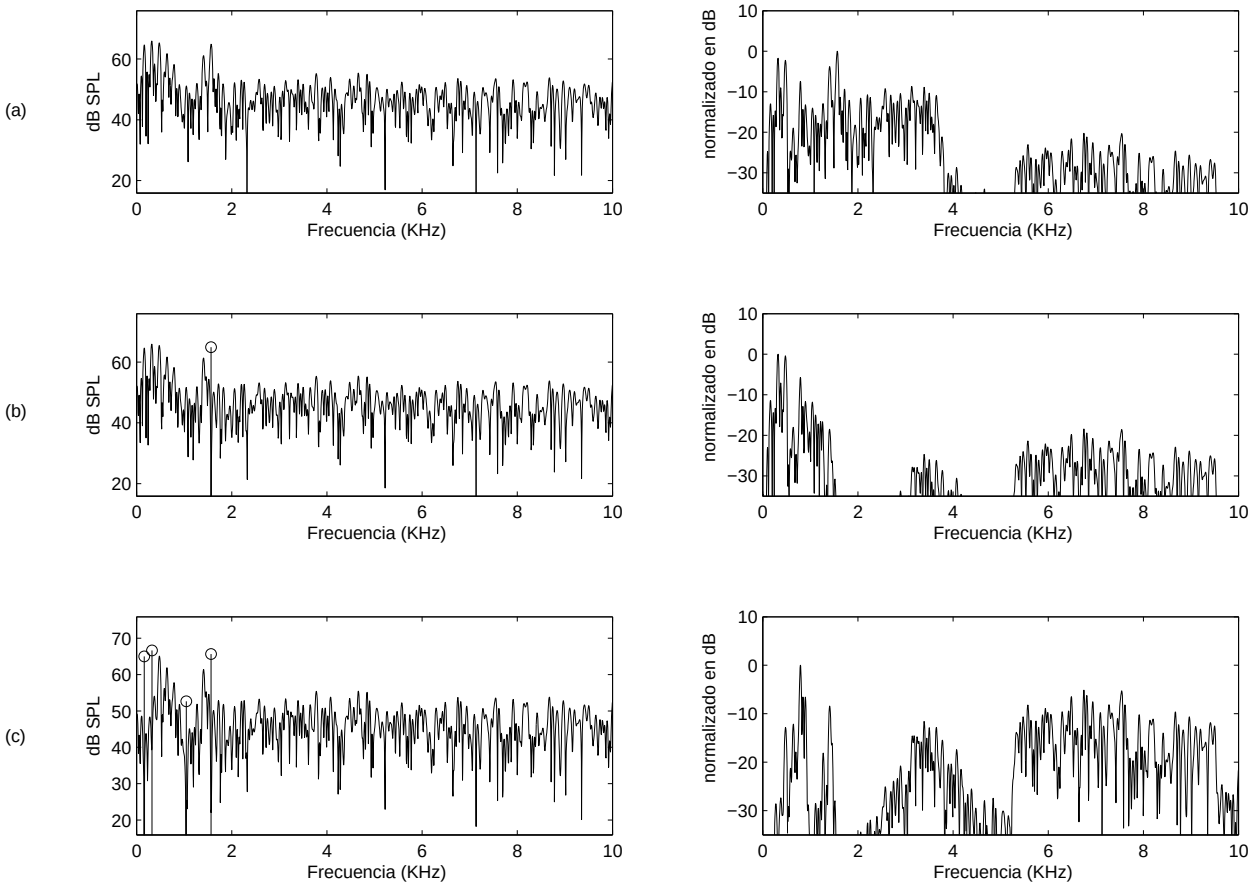

Figura 5.16: *Ejemplo de funcionamiento de la medida PMP para el caso una señal vocal sonora más ruido. (a) Espectro de energía de la señal de entrada (primera columna) y medida perceptual inicial (segunda columna), (b) espectro de energía del residuo (se indican los parámetros del tono extraído usando una recta terminada en círculo) y medida perceptual PMP para la iteración $i = 2$, (c) iteración $i = 5$.*

Cuadro 5.2: *Preferencia en (%) de PMP (banda de Bark) sobre PAMP (frecuencia) cuando se aplica un modelo tonal con 25 tonos por segmento.*

Señal	Preferencia (%)
Suzanne Vega	70
German male speech	100
English female speech	90
Harpsichord	100
Castanets	60
Pitch pipe	70
Bagpipes	70
Glockenspiel	60
Plucked strings	100
Trumpet solo	70
Orchestra piece	90
Contemporary pop	100

Para verificar la propuesta de medida perceptual en banda de bark (PMP), se ha realizado un experimento práctico para comparar la calidad subjetiva del modelo tonal implementado con las dos opciones analizadas: en frecuencia y en banda de Bark. La calidad perceptual evaluada de forma subjetiva para ambas definiciones de medida perceptual se compara en la tabla 5.2. Para obtener esta tabla, cada segmento de la señal de entrada se modela mediante los 25 primeros tonos extraídos con ambos enfoques. Se han elegido para realizar el experimento algunas de las señales de prueba recomendadas por el grupo MPEG [MPEG01] con calidad CD mono. Para esta prueba, se ha implementado un modelo tonal usando segmentos de 23-ms y ventanas de Hamming en análisis con un solapamiento entre tramas del 50 %. Se han realizado unos tests de audición usando la metodología del triple estímulo con referencia ciega, en la que tres señales O, A y B se han presentado a diez personas con experiencia en la evaluación de la calidad de señales de audio. La señal inicial, O, es siempre el original, mientras que las señales A y B son las señales resultantes del modelo tonal bajo PMP y PAMP, respectivamente. Se presentan al evaluador tres veces cada señal consecutivamente, aunque el orden de A o B es aleatorio. Se pregunta al oyente qué señal (A o B) es más parecida al original. Los resultados en media de todos los oyentes al modelar 25 tonos con cada método se presentan en la tabla 5.2.

Como se observa, la ventaja de la medida perceptual en banda de Bark (PMP) es prácticamente generalizada sobre la medida en frecuencia (PAMP). La causa hay que buscarla en los tonos de alta frecuencia que se extraen en el caso de PAMP, los cuales producen pitidos de alta frecuencia molestos para el oyente. Sin embargo, para el caso de PMP, al ser más robusto frente a estos errores se genera una señal de mayor calidad. En base a estos resultados, se utilizará como herramienta para implementar el modelo tonal el método *matching pursuits* guiado perceptualmente mediante una medida psicoacústica definida en banda de Bark (o *Perceptual Matching Pursuits*).

5.3. Estrategias de cuantificación

El modelo sinusoidal ha sido la primera herramienta de análisis paramétrico que se ha introducido en la codificación de audio [George92] [Goodwin98]. Debido a esta temprana adopción del modelo tonal, en la literatura aparecen una serie de esquemas que permiten cuantificar los parámetros del modelo teniendo en cuenta criterios psicoacústicos. La mayoría de estas formas de cuantificación hacen uso de la redundancia en los parámetros del modelo entre tramas adyacentes. Como el modelo tonal extrae la parte estacionaria de la señal de audio, la redundancia entre tramas es enorme y, aplicando este principio, es posible cuantificar los parámetros con un bajo régimen binario.

En la mayoría de las aplicaciones del modelo tonal para codificación de audio, la forma de aprovechar esta redundancia es la codificación diferencial inter-trama de los parámetros tonales con una interpolación en síntesis de estos parámetros [Ali95] [Levine98] [Verma99] [Myburg01]. El modelado tonal, implementado de esta forma, realiza el análisis con el objetivo de crear una serie de caminos tonales o trayectorias que indican la duración de un tono en la señal. La trayectoria de un tono se mantiene de una trama a la siguiente si su frecuencia y amplitud no difieren demasiado de los valores estimados en la trama anterior. Cuando los parámetros de un tono no es posible relacionarlos con los de otro de la trama anterior se crea una nueva trayectoria, aunque este caso es minoritario en las señales de audio. Con esta premisa, es posible implementar una codificación diferencial de los parámetros del modelo tonal: amplitud, fase y frecuencia de cada tono en cada trama.

Sin embargo, al aplicar una codificación diferencial entre tramas, aparecen una serie de problemas que es preciso evitar, si se quiere que el codificador funcione en una aplicación de *streaming* por Internet:

- Al relacionar la información de una trama con la de tramas anteriores, la pérdida de una trama en la transmisión provoca un fuerte impacto en la calidad de la señal decodificada, ya que este error se puede propagar a un gran número de tramas. Así pues, la codificación inter-trama hace al codificador muy sensible frente a los errores en transmisión.

- Si se quiere potenciar la editabilidad, es decir, la posibilidad de comenzar la decodificación en cualquier momento, es preciso evitar en la manera de lo posible la codificación inter-trama, al depender los datos de una trama de los de tramas anteriores, que para esta aplicación en concreto no se conocen.

Además, cuando se utiliza como herramienta de análisis el método *matching pursuits*, se ha demostrado que es mejor evitar la interpolación entre parámetros, necesaria por otra parte para aplicar la codificación diferencial inter-trama.

Si bien es posible mejorar la resistencia a errores y la editabilidad, marcando un tiempo máximo para la codificación diferencial y codificando de forma absoluta la información cada cierto número de tramas, en esta tesis se va a evitar este tipo de codificación. La idea es buscar nuevas soluciones para la codificación, de forma que sólo se utilice en la codificación información intra-trama, lo que viene a significar hacer uso de la redundancia que aparece entre los parámetros del modelo tonal en la trama actual. Se va a tratar por separado la codificación de diferentes tipos de parámetros del modelo tonal.

5.3.1. Cuantificación de la frecuencia

El oído humano es muy sensible a la variación de la frecuencia de un tono, por lo que la frecuencia, vista como parámetro del modelo tonal, requiere un gran gasto en régimen binario para su codificación. Para cuantificar la frecuencia sin que el oído distinga la diferencia, hay que tener en cuenta la sensibilidad del oído medida como JND (*Just Noticeable Difference*) [Zwicker90]. Para el caso de la frecuencia de un tono, el valor de JND en frecuencia depende de la duración del mismo: $0,2$ Bark para 10 ms y $0,01$ Bark para 500 ms [Zwicker90]. Como se observa, la sensibilidad del oído depende de la escala en banda de Bark, la cual es una escala de carácter logarítmico. Por lo tanto, es necesario cuantificar de manera más fina las bajas frecuencias, mientras que se puede relajar la cuantificación de las altas frecuencias.

Hay que tener en cuenta que las frecuencias extraídas por el método MP deben tener carácter lineal, ya que de esta forma se hace uso del algoritmo FFT para su implementación eficiente. Así pues, en la cuantificación de las frecuencias es necesario pasar de una distribución lineal de las frecuencias a una distribución logarítmica. Una forma de realizar este cometido es la estrategia de cuantificación de frecuencias adoptada en [Ali95]. El autor propone dividir las frecuencias lineales en cuatro grupos. Así, suponiendo una frecuencia de muestreo de la señal de audio analógica igual a 44,1 kHz, este esquema propone dividir el eje de frecuencias en:

- Grupo 1: de 0 a 2,75 kHz.
- Grupo 2: de 2,75 kHz a 5,5 kHz.
- Grupo 3: de 5,5 kHz a 11 kHz.
- Grupo 4: de 11 kHz a 22 kHz.

De esta forma, se utiliza el doble del número de escalones de cuantificación en el primer grupo que en el resto. En concreto, en [Ali95], se utilizan 256 escalones de cuantificación para el primer grupo y 128 para el resto. Como resultado, se tiene una separación en frecuencia de 11 Hz para el grupo 1 (bajas frecuencias) que corresponde a $0,107$ Bark. Notar que es preciso indicar al decodificador el grupo al que corresponde la frecuencia actual (2 bits) y el escalón de cuantificación dentro del grupo (8 bits para grupo 1 y 7 bits para el resto de grupos). Es posible incrementar la sensibilidad del esquema simplemente incrementando el número de escalones de cuantificación para cada grupo. En principio, al no ser posible conocer la duración de cada tono en codificación, es necesario tomar un valor de compromiso.

En lo relativo a la redundancia de información intra-trama, la única herramienta que es posible utilizar es la detección de frecuencias que estén armónicamente relacionadas. Si en la trama actual hay un conjunto armónico, es posible sustituir la cuantificación de forma individual de todas estas frecuencias por la transmisión de la frecuencia del tono fundamental y el número de frecuencias armónicas implicadas [Myburg04]. Sin embargo, este enfoque es de difícil aplicación, debido al efecto conocido como *inharmonicity* (o relación no entera de las frecuencias de un complejo armónico) y a la posibilidad en la práctica de que exista más de un complejo armónico simultáneamente. En esta tesis no se ha tenido en cuenta, por lo tanto, esta redundancia, aunque es una posibilidad abierta a la hora de reducir la cantidad de información correspondiente a las frecuencias con carácter intra-trama.

5.3.2. Cuantificación de la fase

Aunque el oído no es sensible a la fase para un tono simple o un par de tonos, es capaz de distinguir las diferencias de fase para tres o más tonos presentes en la señal de audio [Zwicker90]. Esta afirmación, derivada de estudios psicoacústicos, implica que la fase debe ser cuantificada con precisión. Así, en [Ali95], se utiliza un cuantificador uniforme de 6 bits para la cuantificación de la fase.

Cuando se utiliza un enfoque de codificación inter-trama, se han desarrollado estrategias de cuantificación de las fases que consiguen reducir de manera notable la cantidad de recursos binarios dedicados a este parámetro del modelo tonal. Por ejemplo, en [Levine98b], no se envía la fase, salvo cuando un segmento de audio es etiquetado como transitorio. Si esto no es así, se aplica una reconstrucción sin envío de la fase en el modelo tonal, usando un algoritmo que evita discontinuidades de fase basado en la interpolación cúbica de la misma. Sin embargo, la calidad de audio conseguida dista mucho de ser transparente, afirmando algunos autores [Ali95] [Jensen02] que es necesario enviar la fase para conseguir alta calidad.

En esta tesis se ha utilizado la codificación uniforme de todas las fases como método de cuantificación. Este enfoque se ha tomado ante la imposibilidad de utilizar información psicoacústica relativa a la fase. Sin embargo, una línea de futuro se abre ante la posibilidad de discriminar qué tonos de los perceptualmente importantes son sensibles a la variación de la fase y cuáles no. Parece lógico pensar que estos tonos serán aquellos con mayor importancia perceptual, los cuales pueden ser discriminados gracias al guiado perceptual del método MP. Un mayor estudio se necesita realizar en esta tarea.

5.3.3. Cuantificación de la amplitud

Una vez tratados los problemas relacionados con la cuantificación de frecuencia y fase como parámetros del modelo tonal, se pasa a considerar la amplitud de los tonos. En este caso, se cuenta con una ventaja, que es la posible utilización de información perceptual para la cuantificación de estos valores. Como es lógico, la cuantificación de las amplitudes es un tema ya abordado en la bibliografía, aunque el enfoque comúnmente adoptado se basa en la codificación diferencial inter-trama.

Si se desea utilizar la información perceptual derivada de un modelo de enmascaramiento, lo primero a realizar es calcular la máscara de tonos de la trama actual y eliminar los tonos no audibles (por debajo de la máscara). En general, la amplitud de cada tono debería ser cuantificada de forma que el error de cuantificación esté lo más cercano posible, pero siendo menor, que el umbral de enmascaramiento estimado. Con esta premisa, se conseguirá asegurar que la señal codificada sea perceptualmente idéntica a la original, sacando así máximo provecho del fenómeno de enmascaramiento simultáneo del sistema auditivo humano. El inconveniente de este enfoque reside en que resulta un número variable de bits por cada amplitud tonal, lo que conlleva informar al decodificador del número de bits asignados a cada amplitud. Además, el decodificador necesitará el valor máximo y mínimo de cada cuantificador. Al final, este enfoque de codificación intra-trama (aprovechando la máscara simultánea generada por la señal de audio en la trama actual) ha sido descartado en la literatura por la gran cantidad de información lateral que es preciso enviar al decodificador.

Así, para evitar el envío de información lateral, la estrategia de cuantificación establecida

para la codificación de amplitudes, en aquellos codificadores de audio que incluyen un modelado sinusoidal [Ali95] [Levine98] [Brinker02], utiliza un escalón fijo de cuantificación para todos los tonos. Si este escalón de cuantificación se elige para lograr la transparencia perceptual del tono más crítico (el que necesita más bits), el régimen binario resultante será muy elevado. Como consecuencia, en la mayoría de las situaciones se emplea un escalón de cuantificación bastante grande, provocando que la calidad de la señal codificada no sea transparente en absoluto. Por ejemplo, en [Ali95], se utilizan 10 bits para la codificación de las amplitudes de las nuevas frecuencias (aquellas que comienzan un camino tonal), lo que asegura, según el autor, que sólo el 2 % de los tonos no consiguen una codificación transparente. Para las frecuencias que continúan un camino tonal desde la trama anterior se utiliza cuantificación diferencial y, también, se necesitan 10 bits para el mismo requisito de codificación transparente.

En esta tesis, el estudio de cuantificación de las amplitudes se ha centrado en el análisis de la información intra-trama, por lo que se utiliza el umbral de enmascaramiento simultáneo. La idea se basa en la utilización de una adaptación hacia atrás [Rodrigues00], de forma que codificador y decodificador trabajen en los mismos valores, y se evite el envío de información lateral, a pesar de usar un tamaño de escalón de cuantificación variable para cada amplitud. El algoritmo que se propone [Vera04b] reduce drásticamente el envío de información lateral, a costa de incrementar un poco la complejidad del decodificador. El precio a pagar es el cálculo de unos sencillos umbrales de enmascaramiento para la cuantificación de las amplitudes, tanto en el codificador como en el decodificador, que aseguren la codificación trasparente con un bajo consumo de recursos binarios.

Para que la estrategia de cuantificación de las amplitudes funcione, ésta debe permitir el cálculo, tanto en codificación como en decodificación, del número de bits por amplitud bajo un criterio de transparencia perceptual, así como la obtención de los valores máximo y mínimo de cada cuantificador. Sin embargo, un requisito a priori es el cálculo en decodificación del umbral de enmascaramiento. Seguidamente, se explica qué umbral de enmascaramiento es posible calcular en el decodificador. A partir de ahora, el índice i representará el i-ésimo tono para cuantificar. El umbral de enmascaramiento individual que genera el tono i-ésimo (una vez enviado) en la banda de bark b sobre el resto de tonos, se puede calcular a partir de la expresión (5.23), propuesta en [Ali95],

$$T_i(b)(\text{dB SPL}) = \begin{cases} A_i - 14{,}5 - b - 31(b_i - b), & b < b_i \\ A_i - 14{,}5 - b, & b = b_i \\ A_i - 14{,}5 - b - \left(22 + min\left(\frac{230}{f_i}, 10\right) - 0{,}2A_i\right)(b_i - b), & b > b_i \end{cases} \tag{5.23}$$

donde b_i, A_i y f_i representan el índice en banda de Bark, la amplitud (expresada en dB SPL) y la frecuencia (expresada en Hz) correspondientes al i-ésimo tono audible, respectivamente. Por lo tanto, para calcular este umbral se debe suponer que ya se han enviado las frecuencias de cada tono.

Como el objetivo es no enviar información lateral, el umbral de enmascaramiento individual se debe obtener tanto en el codificador como en el decodificador, a partir de amplitudes cuantificadas [Rodrigues00]. Una forma de implementar esta limitación, utilizando un umbral de enmascaramiento conservador, viene dada por la expresión (5.24), modificada a partir de la anterior,

$$T_{q_i}(b)(\text{dB SPL}) = \begin{cases} Q_i - \frac{\Delta_i}{2} - 14{,}5 - b - 31(b_i - b), & b < b_i \\ Q_i - \frac{\Delta_i}{2} - 14{,}5 - b, & b = b_i \\ Q_i - \frac{\Delta_i}{2} - 14{,}5 - b - \big(22 + min\big(\frac{230}{f_{q_i}}, 10\big) - 0{,}2Q_i\big)(b_i - b), & b > b_i \end{cases} \tag{5.24}$$

donde Q_i, Δ_i y f_{q_i} representan la amplitud cuantificada (en dB SPL), el tamaño del escalón de cuantificación (en dB) y la frecuencia cuantificada (en Hz) correspondientes al i-ésimo tono audible, respectivamente.

El umbral de enmascaramiento compuesto para cuantificación de las amplitudes C_i, una vez se ha enviado el tono i-ésimo, se compone a partir del umbral compuesto anterior, C_{i-1}, y del umbral del tono actual, T_i, de la forma,

$$C_i(b)(\text{dB SPL}) = max\big(T_{q_i}(b), C_{i-1}(b)\big) \tag{5.25}$$

El problema de inicializar el umbral de enmascaramiento compuesto C_0 depende de la información que tenga el decodificador. En principio, el único umbral que es posible inicializar en el decodificador es el umbral de silencio. Sin embargo, en el caso de un codificador completamente paramétrico, si el decodificador tiene la información de la potencia de ruido (por bandas en la trama actual), puede generar el umbral NMT y añadirlo al umbral de silencio. Por simplicidad, se supondrá que el umbral C_0 se inicializa al umbral de silencio. Como se puede deducir de (5.25), en el umbral de enmascaramiento compuesto se incluye la interacción entre los umbrales individuales de los tonos en las sucesivas iteraciones.

En [Zwicker90], el umbral de enmascaramiento compuesto se estima usando el operador suma sobre los valores lineales (no en dB SPL). Aunque el umbral así calculado da lugar a valores menos conservadores que los obtenidos con el máximo en la ecuación (5.25), el uso del máximo se justifica por la reducción de la complejidad conseguida, al evitar la conversión de valores dB SPL a unidades lineales. Como contrapartida, se obtiene una menor eficiencia de codificación. Como se puede observar, los umbrales de enmascaramiento se calculan en la escala de banda de Bark.

Una vez explicado cómo se obtienen los umbrales en codificador y decodificador, es necesario determinar los valores de configuración del cuantificador para cada amplitud. Estos valores de configuración se tienen que conocer antes de cuantificar sin enviar información lateral. Los valores de configuración para el cuantificador del tono i-ésimo son el rango dinámico de la amplitud (v_{min_i} y v_{max_i}), y la mínima relación señal a ruido (SNR) necesaria para lograr codificación trasparente, que se denotará como R_i a partir de ahora. Estos valores, que variarán de un tono a otro, se deben calcular teniendo en cuenta el umbral de enmascaramiento compuesto.

El proceso que se sigue en el algoritmo propuesto para el cálculo de los valores de configuración de cada cuantificador se explica a continuación. En un codificador perceptual de audio, la SNR necesaria R_i para cuantificar el tono i-ésimo, de forma que el error de cuantificación no sea audible, se debe obtener como la diferencia entre la amplitud del tono A_i y el umbral de enmascaramiento compuesto evaluado en la banda de Bark b_i del tono actual. Bajo este enfoque, en este esquema de cuantificación, la SNR requerida R_i se debería calcular como $A_i - C_i(b_i)$. Sin embargo, esta aproximación no se puede utilizar porque el decodificador desconoce la amplitud del

tono actual A_i. Para resolver este inconveniente, se propone el uso del umbral de enmascaramiento individual que aparece en la ecuación (5.23), en vez de usar el umbral de enmascaramiento compuesto. Con esta solución, la SNR R_i se calcula como,

$$R_i(\text{dB}) = A_i - T_i(b_i) = 14{,}5 + b_i \tag{5.26}$$

Como se observa en la ecuación, al depender el umbral de enmascaramiento individual de la amplitud del tono A_i, la SNR R_i sólo depende de la banda de bark b_i del tono i-ésimo. Con esta elección del valor de la SNR, R_i, se asegura la transparencia perceptual de la codificación y su valor no depende de la amplitud del tono A_i.

El siguiente valor a fijar es el valor mínimo del cuantificador v_{min_i} para el i-ésimo tono audible. Este parámetro se puede igualar directamente al umbral de enmascaramiento compuesto obtenido antes de cuantificar el tono actual, C_{i-1}, que evaluado en la banda de bark del tono actual queda,

$$v_{min_i}(\text{dB SPL}) = C_{i-1}(b_i) \tag{5.27}$$

Este valor se justifica debido a que si el tono es audible su amplitud será siempre mayor que el umbral de enmascaramiento.

Para finalizar, queda por determinar el valor máximo v_{max_i} del cuantificador de la amplitud del tono i-ésimo. En principio, para este propósito se puede aplicar el máximo valor posible de la amplitud de un tono, que es de 96 dB SPL. Sin embargo, la inmensa mayoría de las amplitudes de los tonos son mucho menores que este valor. La solución es emplear el valor de 96 dB SPL para el primer tono y tomar la amplitud de este tono como el valor máximo del cuantificador para los demás. Esto se podrá realizar siempre que se envíe, en primer lugar, el tono de mayor amplitud al decodificador. El valor máximo de los cuantificadores queda,

$$v_{max_i}(\text{dB SPL}) = \begin{cases} 96 & i = 1 \\ Q_1 & i > 1 \end{cases} \tag{5.28}$$

donde Q_1 es la amplitud cuantificada del primer tono. Esta selección del valor máximo obliga a cuantificar primero el tono de mayor amplitud.

A partir de la expresión (5.28), hay que tener en cuenta que el primer tono debe ser cuantificado con mucha precisión para evitar que se propague el error de cuantificación para el resto de tonos. La forma de evitar este inconveniente consiste en utilizar una SNR para el primer tono, R_1, mucho mayor que la máxima posible SNR para el resto de tonos. Como la máxima banda de Bark para una frecuencia de muestreo de $44,1$ kHz es de $b_i|_{max} = 26$, la máxima SNR para el resto de tonos será: $R_i|_{max} = 14{,}5 + b_i|_{max} = 14{,}5 + 26 = 40{,}5$ dB, según la ecuación (5.26). Basándose en este resultado, se elige para el primer tono una SNR mucho mayor que la máxima posible, quedando $R_1 = R_i|_{max} + 10\text{dB} = 50{,}5$ dB.

Una vez conocidos los valores de configuración del cuantificador i-ésimo, es posible determinar el tamaño del escalón de cuantificación Δ_i y, por consiguiente, el número de bits n_i asignados a la amplitud de tono i-ésimo. Estos valores se van a determinar teniendo en cuenta que la cuantificación es uniforme y que se trabaja directamente con las amplitudes expresadas en dB SPL.

El tamaño del escalón de cuantificación Δ_i se calcula para obtener, al menos, la relación señal a ruido R_i entre la amplitud del tono actual A_i y la amplitud de la señal de error e_i. El máximo

error en la amplitud $e_i|_{max}$ se obtiene cuando la amplitud cuantificada del tono actual Q_i toma el máximo error de cuantificación:

$$Q_i|_{max}(\text{dB SPL}) = A_i(\text{dB SPL}) + \frac{\Delta_i}{2}(\text{dB}) \tag{5.29}$$

El máximo error se expresa en dB SPL como la diferencia entre la amplitud cuantificada con máximo error de cuantificación y la amplitud original:

$$e_i|_{max}(\text{dB SPL}) = 20 \cdot log_{10}\Big(10^{\frac{|Q_i|_{max}(\text{dB SPL})}{20}} - 10^{\frac{A_i(\text{dB SPL})}{20}}\Big) \tag{5.30}$$

Según (5.29), la expresión (5.30) se puede simplificar de la forma:

$$e_i|_{max}(\text{dB SPL}) = A_i(\text{dB SPL}) + 20 \cdot log_{10}\Big(10^{\frac{\Delta_i/2(\text{dB})}{20}} - 1\Big) \tag{5.31}$$

La relación señal a ruido que se obtiene cuando el error es máximo $e_i|_{max}$ es el mínimo valor posible de SNR. Sustituyendo este mínimo valor por R_i, para conseguir al menos este valor de SNR, el máximo error de la amplitud $e_i|_{max}$ se puede expresar en dB como:

$$e_i|_{max}(\text{dB SPL}) = A_i(\text{dB SPL}) - R_i(\text{dB}) \tag{5.32}$$

Teniendo en cuenta la ecuación (5.31), la expresión (5.32) se puede simplificar de la forma:

$$R_i(\text{dB}) = -20 \cdot log_{10}\Big(10^{\frac{\Delta_i/2(\text{dB})}{20}} - 1\Big) \tag{5.33}$$

Finalmente, de acuerdo a la ecuación (5.33), el tamaño del escalón de cuantificación Δ_i se puede calcular a partir de:

$$\frac{\Delta_i}{2}(\text{dB}) = 20 \cdot \log_{10}\left(1 + 10^{-\frac{R_i(\text{dB})}{20}}\right) \tag{5.34}$$

El número de bits asignados al tono actual, n_i, se obtiene a partir del valor del tamaño del escalón de cuantificación Δ_i y de los valores de configuración del cuantificador del tono i-ésimo como:

$$n_i = \left\lceil \log_2\left(\frac{v_{max_i}(\text{dB SPL}) - v_{min_i}(\text{dB SPL})}{\Delta_i(\text{dB})}\right)\right\rceil \tag{5.35}$$

Todo lo expresado hasta ahora se escribe de forma algorítmica a continuación. Se expresa detalladamente la obtención de los valores de configuración de los cuantificadores de cada amplitud:

1. Se inicializa el umbral de enmascaramiento C_0 al umbral de silencio.
2. Para el primer tono, $i = 1$, que es el tono de máxima amplitud, entonces:
   a) Los valores de configuración del primer cuantificador son:
   - $v_{max_1} = 96$ dB SPL (el máximo valor posible); $v_{min_1} = C_0(b_1)$.
   - $R_1 = |R_i|_{max} + 10 = 50{,}5$ dB, ya que Q_1 se usa más tarde como valor máximo para el resto de cuantificadores.

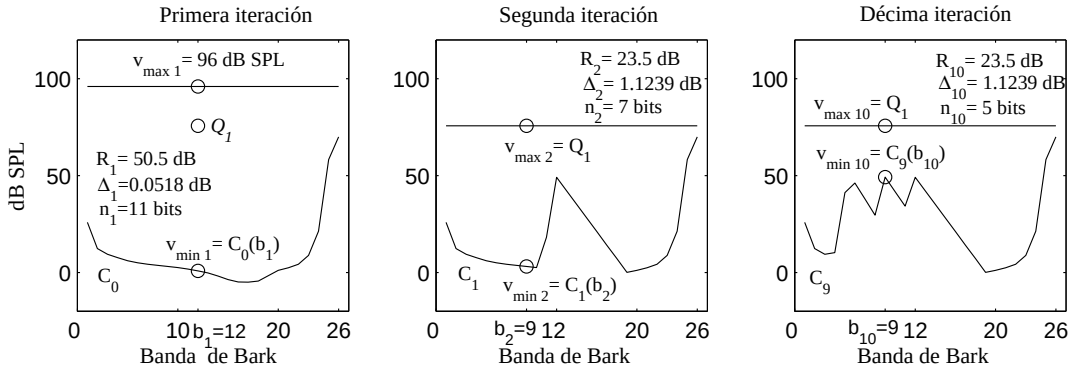

Figura 5.17: *Ejemplo de funcionamiento del algoritmo propuesto para la cuantificación de las amplitudes.*

b) Se calcula el escalón Δ_1 mediante (5.34), y el número de bits n_1 a partir de (5.35), y se cuantifica A_1 obteniendo la amplitud cuantificada Q_1.

c) El umbral de enmascaramiento C_1 se compone con el umbral individual del primer tono mediante (5.24) y (5.25).

3. Para $i = 2, \ldots, K$, siendo K el número de tonos audible, entonces:

a) Los valores de configuración de los cuantificadores son:

- $v_{max_i} = Q_1$; $v_{min_i} = C_{i-1}(b_i)$.
- $R_i = 14{,}5 + b_i$.

b) Se calcula el escalón Δ_i mediante (5.34), y el número de bits n_i a partir de (5.35), y se cuantifica A_i obteniendo la amplitud cuantificada Q_i.

c) El umbral de enmascaramiento C_i se compone con el umbral individual del tono actual mediante (5.24) y (5.25).

Una ilustración del funcionamiento del algoritmo se presenta en la figura 5.17. En esta figura, la línea superior representa el valor máximo de cada cuantificador, mientras que la inferior el valor mínimo en cada banda de bark. Como se observa, este valor mínimo siempre coincide con el umbral de enmascaramiento compuesto en la banda de bark b_i del tono actual. En la primera iteración, el número de bits asignado al primer tono, $n_1 = 11$ bits, es mayor que para el resto puesto que tanto $v_{max_1} = 96dB\ SPL$ como $R_1 = 50,5dB$ son especialmente grandes sólo para esta primera iteración. Así, el número de bits asignados para el resto de tonos es menor que n_1 debido a: 1) v_{max_i} se fija a Q_1 y 2) R_i se obtiene a partir del umbral de enmascaramiento individual de cada tono. Aún más, el número de bits n_i tiende a ser menor conforme el índice i aumenta, porque v_{min_i} tiende a crecer al aumentar el umbral de enmascaramiento compuesto.

El algoritmo que se acaba de explicar es preciso aplicarlo tanto al codificador como al decodificador para evitar el envío de información lateral. En realidad, el codificador debe enviar el número de tonos audibles como información lateral para que el algoritmo funcione. Además, es preciso tener en cuenta que se ha supuesto que ya se conocen, cuando se ejecuta este algoritmo, las frecuencias de los tonos en el decodificador. Es importante reseñar que, como el algoritmo es muy sensible a errores de cuantificación, parece lógico protegerlo de alguna forma ante este

problema. Una manera simple de realizar este cometido es enviando al decodificador el número de bits resultante de la cuantificación de las amplitudes. Con este valor, se evita la propagación de un error de transmisión al resto de información del codificador (por ejemplo a otras tramas de señal codificada).

El siguiente paso será comparar los resultados del algoritmo propuesto con otras opciones presentes en la literatura, con el objetivo de mostrar su buen funcionamiento. Para ello se ha utilizado el esquema experimental mostrado en la figura 5.4. La extracción tonal se ha implementado mediante *matching pursuit* (guiado por energía) con ventana rectangular en análisis y trapezoidal en síntesis. Como modelo perceptual se ha usado el descrito en [MPEG92]. Con este esquema, se pretende verificar tanto el régimen binario obtenido como la calidad subjetiva del algoritmo propuesto para la cuantificación de las amplitudes.

En la figura 5.18 se comparan los resultados en régimen binario del algoritmo propuesto y del algoritmo presentado en [Ali95] para cuantificar las amplitudes, el cual siempre utiliza el mismo número de bits (10 bits) por amplitud. En la figura 5.18 se representa el régimen binario que se obtiene cuando se varía el número de tonos extraídos según la relación RSR (*Residual to Signal Ratio*). Las señales de prueba utilizada son un subconjunto de las propuestas por MPEG, donde se incluyen señales muy tonales (ver tabla 5.1). En esta figura se observa que la mejora en régimen binario del método propuesto es tanto mayor cuanto menor es el valor de RSR. Esto es debido a las siguientes razones: 1) Los primeros tonos extraídos son normalmente aquellos más relevantes perceptualmente, y 2) El esquema de cuantificación de amplitudes propuesto asigna un número variable de bits a cada tono dependiendo de su importancia perceptual. Cabe destacar que el método propuesto consigue un régimen binario muy bajo para la cuantificación de las amplitudes de los tonos (menor de $0,16$ bits/muestra en media).

Para verificar el correcto funcionamiento del algoritmo propuesto se han llevado a cabo unos tests psicoacústicos. La señal evaluada ha sido la obtenida al sumar la señal cuantificada del modelo tonal con el residuo, de acuerdo con el esquema de la figura 5.4. En la figura 5.19 se comparan los resultados de estos tests medidos en la escala MOS, aplicando por un lado el algoritmo propuesto de codificación de amplitudes y el algoritmo descrito en [Ali95] por otro. Como se observa en la figura, el esquema propuesto logra una alta calidad perceptual, cercana a la transparencia, mientras que el método de [Ali95] da lugar a sonidos metálicos al cuantificar las amplitudes, lo que resulta en una calidad perceptual mucho menor. Este efecto es muy común cuando se utiliza un número fijo de bits por amplitud y este valor no es suficiente para los tonos perceptualmente más importantes. Concretamente, estos tonos son aquellos que el oído detecta con mayor detalle, lo que provoca un derrumbe en la calidad medida en la escala MOS. Para concluir es posible afirmar que el método propuesto para cuantificar las amplitudes, basado en principios psicoacústicos, consigue una alta calidad perceptual en el proceso de cuantificación sin aumentar el régimen binario necesario.

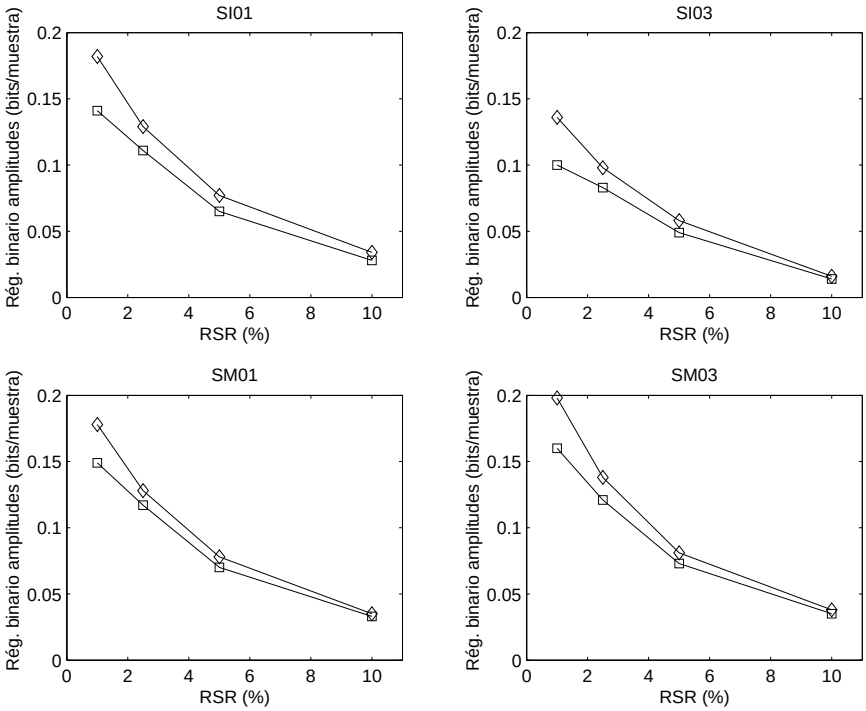

Figura 5.18: *Variación del régimen binario (bits/muestra) en media para la cuantificación de las amplitudes conforme la relación RSR(%) aumenta. Método en [Ali95] (rombos), método propuesto (cuadrados).*

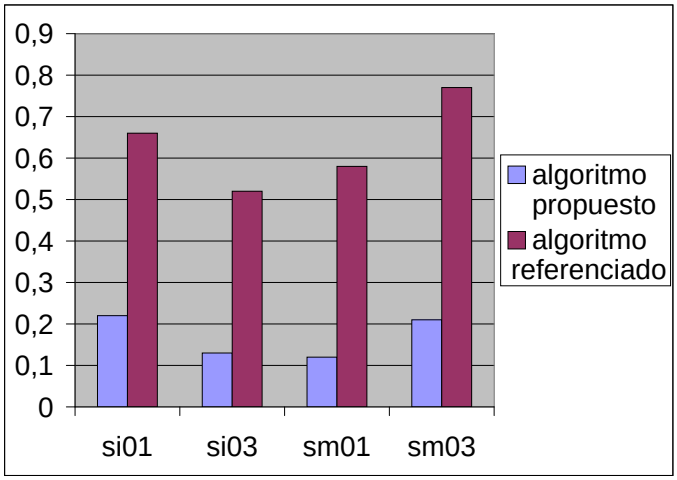

Figura 5.19: *Comparación de resultados subjetivos (valores de ΔMOS) obtenidos por el algoritmo de cuantificación de las amplitudes de los tonos propuesto y por el presentado en [Ali95]*

Capítulo 6

Modelado de transitorios

En relación al modelado de transitorios, el número de trabajos relacionados con el tema que puede encontrarse en la bibliografía especializada es muchísimo menor que en el caso del modelado tonal. La razón hay que buscarla primero en que la presencia de fuertes transitorios en la señal de audio es bastante reducida, por lo que un codificador compuesto simplemente por un modelo tonal más un modelo de ruido es capaz de dar una calidad aceptable para un amplio conjunto de señales de audio. En segundo lugar, el problema de los transitorios reside en que representan un conjunto dispar de comportamientos de la señal. Si en el caso de modelo tonal, la señal a modelizar está formada por tonos estables, en el caso de los transitorios la señal puede estar formada por los incrementos bruscos de energía de una amplia variedad de instrumentos, lo que dificulta el diseño de un modelo simple. Como consecuencia de esta propiedad de los transitorios, el modelo que mejor funciona es un modelo generalista, que pueda caracterizar un amplio rango de comportamientos tiempo-frecuencia. Esta es la causa principal de que el modelo con mayor éxito en la literatura sea el método *matching pursuits* con diccionarios adaptados a las características transitorias de la señal [Goodwin97b] [Nieuwenhuijse98].

Si se consigue un modelo de transitorios con una calidad aceptable, sus aplicaciones no sólo estarán en el ámbito de la codificación, sino que también se podrán realizar interesantes aplicaciones en el campo del tratamiento digital de señales de audio con esta herramienta. Así, si se dispone de un modelo de transitorios, es posible realizar un amplio rango de modificaciones sobre la señal, así como, incluso, clasificar los transitorios de señal o extraer propiedades del ritmo de la misma. Se puede afirmar, por tanto, que una herramienta de modelado de transitorios eficiente puede ser de gran utilidad en múltiples aplicaciones del audio.

6.1. Diccionarios paramétricos con *matching pursuits*

Para el método MP, al igual que para otras descomposiciones atómicas, el diseño del diccionario es la clave para adaptar la descomposición a la aplicación en cuestión. En general, está claro que cuanto mayor es el diccionario, y más comportamientos tiempo-frecuencia se incluyen en su definición, mayor es la capacidad de compresión del método, aunque, como contrapartida, mayor será su complejidad, que crece de forma exponencial con respecto al tamaño del diccionario.

Una característica deseable para un modelo de transitorios es que la definición del diccionario esté basada en parámetros con significado físico. Sólo cuando los parámetros de los átomos tienen

esta propiedad se designa el modelo implementado como modelo paramétrico. En aplicaciones de audio, es muy importante que los átomos del diccionario puedan localizarse en tiempo/frecuencia, a partir de algunos de sus parámetros, porque de esta forma es posible implementar modificaciones de señal sobre los datos codificados, como puede ser, por ejemplo, el cambio de tempo.

6.1.1. Átomos de Gabor

Los átomos localizados en tiempo-frecuencia fueron introducidos por Gabor [Gabor46] y están diseñados de forma que tienen una buena discriminación tanto en tiempo como en la frecuencia. Volviendo a escribir la definición de estos átomos en tiempo discreto según,

$$g_{\{s,\omega,\tau\}}[n] = f_s[n-\tau]e^{j\omega(n-\tau)} \tag{6.1}$$

es posible comprobar como el diccionario de átomos de Gabor es paramétrico, puesto que los parámetros $\{s, \omega, \tau\}$ definen la escala, la frecuencia y la localización en el tiempo, respectivamente.

El principal problema de los átomos de Gabor es el tamaño del diccionario necesario para incluir un conjunto aceptable de comportamientos tiempo-frecuencia en la señal. Como consecuencia, la complejidad para desarrollar este modelo es prohibitiva en aplicaciones de tiempo real. Además, para implementar un modelo de transitorios, estos átomos tienen el inconveniente de presentar simetría par, por lo que es imposible modelizar convenientemente un transitorio. Como se observó en la figura 4.15, estos átomos producen un pre-eco considerable cuando intentan modelizar transitorios.

Sin embargo, los transitorios reales no son tan simples como los de la figura 4.15, sino que son señales mucho más complicadas de modelizar. Para mostrar este hecho, se incluye la figura 6.1, donde se modela un transitorio de audio con átomos de Gabor. Se puede apreciar en la figura cómo en las primeras iteraciones se modela una señal suave, que se refina en posteriores iteraciones. El inconveniente principal de este enfoque es que al principio no se modela una señal transitoria, sino la forma suave en general de la señal.

6.1.2. Sinusoides amortiguadas exponencialmente

Una forma de evitar el problema de la simetría par de los átomos de Gabor es redefiniendo los átomos de forma que no posean esta característica. La forma más usual encontrada en la bibliografía de realizar este cambio es proporcionar a los átomos de una caída exponencial mediante sinusoides amortiguadas exponencialmente (*Exponentially Damped Sinusoids*, EDS). Ahora la definición de los átomos es más directa,

$$g_{\{a,\omega,\tau\}}[n] = S_a a^{(n-\tau)} e^{j\omega(n-\tau)} u[n-\tau] \tag{6.2}$$

quedando éstos definidos a partir de la terna $\{a, \omega, \tau\}$, que representa el factor de amortiguamiento, la frecuencia y la localización temporal, respectivamente. La ventaja de estos parámetros aparece en dos sentidos. Por un lado, la localización temporal se refiere exactamente al comienzo del transitorio, y por otro, el factor de amortiguamiento define la caída exponencial de la señal. Es importante tener en cuenta que, además, estos átomos se corresponden con la respuesta al impulso de filtros lineales definidos por un polo.

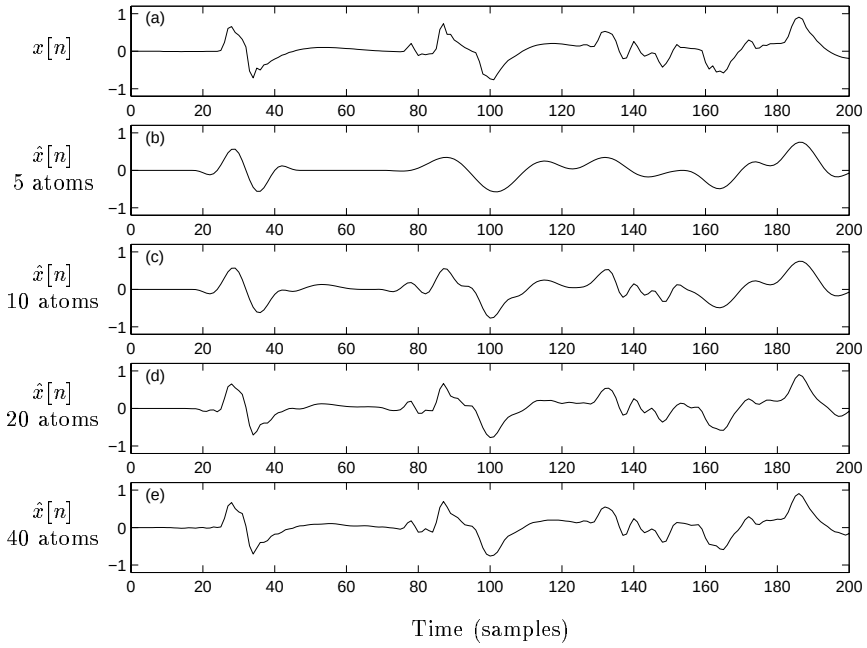

Figura 6.1: *Modelado de señal con MP y átomos de Gabor. La señal es un transitorio de audio de un gong. Se reconstruye la señal modelada con 5, 10, 20 y 40 átomos [Goodwin97].*

La implementación de *matching pursuits* con exponenciales complejas ha sido tratada con detalle en [Goodwin97b]. El problema principal de esta implementación es la enorme cantidad de memoria necesaria para realizar la actualización de correlaciones. Es tal la cantidad de memoria necesaria que en [Goodwin97b] se explica cómo realizar la implementación de MP mediante filtros para intercambiar memoria por cálculo directo de las correlaciones (por multiplicaciones) con ayuda de estos filtros. En realidad, debido a que los átomos EDS son la respuesta al impulso de filtros lineales de un polo, es posible calcular las correlaciones con un banco de filtros de un polo. Una interpretación gráfica de este banco de filtros aparece en la figura 6.2, donde se observan los filtros a implementar con diferentes aproximaciones en el diseño del diccionario. Se puede encontrar en [Goodwin98] un informe detallado de la implementación y la complejidad necesaria asociada.

En la figura 6.3, se representa el análisis realizado con un diccionario EDS del transitorio de audio de un gong. La diferencias sustanciales con el caso del diccionario de átomos de Gabor son dos: 1) en las primeras iteraciones se representan las principales transiciones de la señal y 2) ahora no hay error de pre-eco.

En [Goodwin98], se estudia, además, el error cuadrático medio entre la señal original y la reconstruida en las sucesivas iteraciones, es decir, la potencia de la señal residual. El resultado se representa en la figura 6.4, donde se aprecia que para la señal transitoria usada el diccionario EDS converge más rápidamente que los átomos de Gabor.

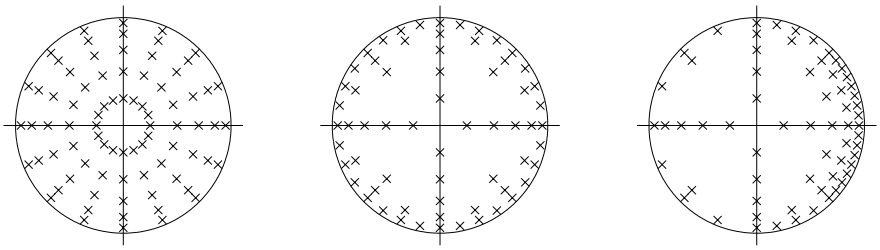

Figura 6.2: *Interpretación mediante bancos de filtros de varias estructuras de diccionario EDS. Los átomos del diccionario se corresponden con la respuesta al impulso de los polos marcados en el plano z.*

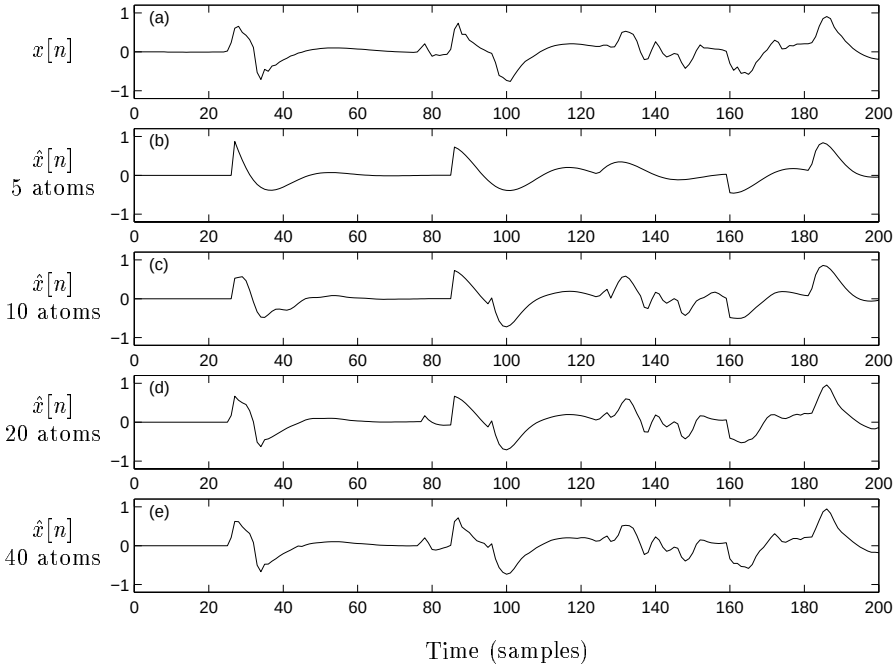

Figura 6.3: *Modelado de señal con MP y diccionario EDS. La señal es un transitorio de audio de un gong. Se reconstruye la señal modelada con 5, 10, 20 y 40 átomos [Goodwin97].*

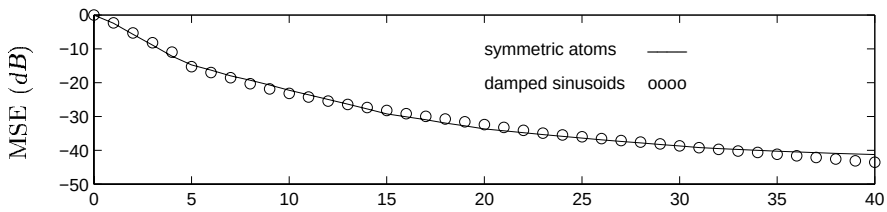

Figura 6.4: *Error cuadrático medio de método MP con átomos de Gabor y exponenciales amortiguadas para un transitorio de audio [Goodwin97].*

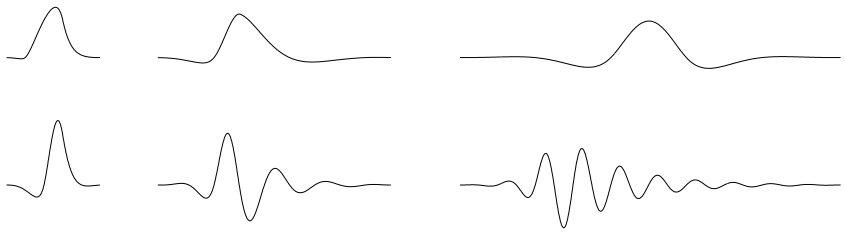

Figura 6.5: *Átomos compuestos variando la frecuencia de modulación y los factores de amortiguamiento.*

6.1.3. Átomos compuestos

Se ha mostrado en las figuras anteriores que tanto el diccionario de átomos de Gabor simétricos como el diccionario EDS no incluyen en su definición la gran cantidad de comportamientos que se pueden encontrar en las señales transitorias de audio. Para señales reales, que pueden ser generadas por sistemas no lineales complejos, parece lógico implementar un diccionario que presente tanto átomos simétricos (Gabor) como asimétricos (EDS). Si se incluyen en un sólo diccionario los átomos de los diccionarios simples, tal y como son, el diccionario resultante se suele llamar diccionario mixto. Sin embargo, este enfoque no es apropiado cuando se utiliza MP, puesto que la actualización de correlaciones se complica, al incluir en el diccionario átomos de diversa naturaleza. Como resultado, todas las correlaciones cruzadas deben ser guardadas en memoria, lo que supone una complejidad excesiva.

Otra alternativa, consiste en redefinir los átomos de forma que incluyan diversos comportamientos [Goodwin98]. Esta forma de diseñar el diccionario lleva a un diccionario compuesto por átomos de diversa naturaleza con una misma definición (*composite atoms*). El enfoque aplicado en [Goodwin98] se basa en considerar átomos compuestos por funciones causales y anticausales, que se pueden definir como,

$$g_{\{a,b,\omega,\tau\}}[n] = S_{a,b}(a^{(n-\tau)}u[n-\tau] + b^{-(n-\tau)}u[-n+\tau] - \delta[n-\tau])e^{j\omega(n-\tau)} \tag{6.3}$$

donde los parámetros de definición son $\{a, b, \omega, \tau\}$: el factor de amortiguamiento causal, el factor de amortiguamiento anticausal, la frecuencia y la localización temporal, respectivamente. Con esta definición, se pueden encontrar átomos de muy diversa naturaleza, como aparece en la figura 6.5.

A continuación, se muestra en la figura 6.6 el análisis realizado con átomos compuestos para un transitorio de audio. En este caso, es visible cómo los átomos se eligen de forma adaptada a la señal, usando átomos con forma causal, por ejemplo, en el primer transitorio, y con forma simétrica en otros. El resultado es una convergencia mucho más rápida, como aparece en la figura 6.7. Ahora la diferencia es mayor incluso en las primeras iteraciones.

Los átomos compuestos ofrecen la posibilidad de determinar con el método MP qué parte de la señal es transitoria (donde se extraen más átomos causales) y cual no. El precio a pagar es un incremento notable del tamaño del diccionario, que incluye gran número de átomos. El resultado es una complejidad prohibitiva a la hora de implementar MP con estos átomos compuestos.

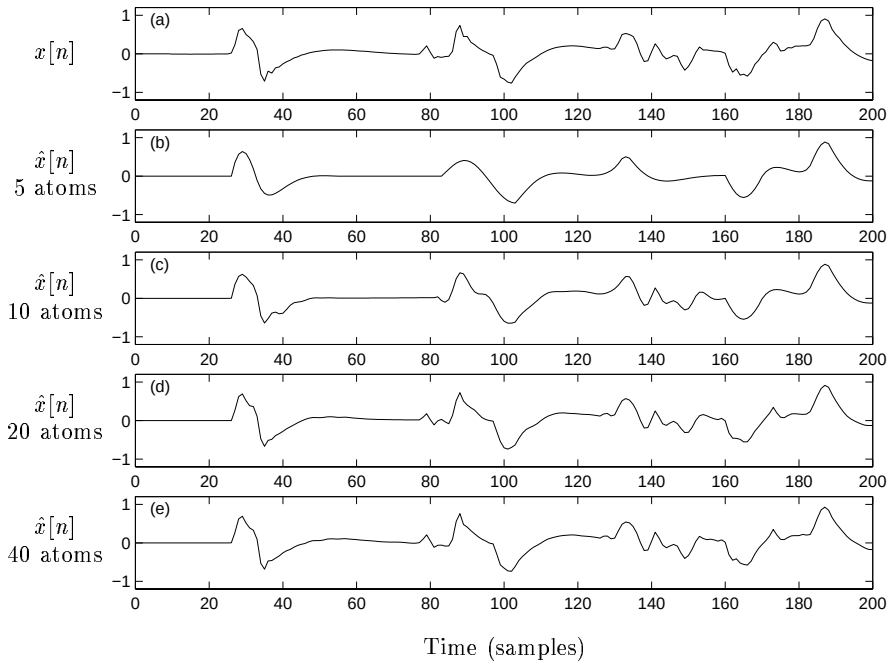

Figura 6.6: *Modelado de señal con MP y diccionario de átomos compuestos. La señal es un transitorio de audio de un gong. Se reconstruye la señal modelada con 5, 10, 20 y 40 átomos [Goodwin97].*

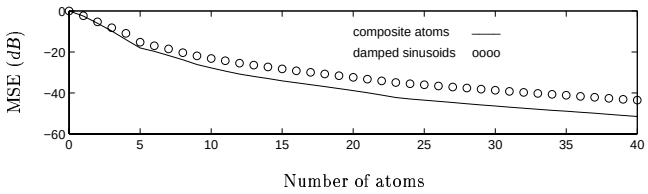

Figura 6.7: *Error cuadrático medio del método MP con exponenciales amortiguadas y átomos compuestos para un transitorio de audio [Goodwin97].*

6.2. Diccionario de funciones wavelet packets

En los diferentes codificadores de audio donde la transformadas wavelets y wavelet-packets se han utilizado, por ejemplo en [Hamdy96] [Ruiz03], los mejores resultados subjetivos corresponden a las señales con mayor número de transitorios. En [Hamdy96], la transformada wavelet se usa para descomponer las partes transitorias de la señal. Si se codifican los valores de esta transformada, la representación obtenida de los transitorios proporciona un régimen binario menor con una calidad más alta que si se utiliza una transformada basada en funciones exponenciales complejas como la DCT. En [Ruiz03], se presenta un codificador de audio basado en transformadas wavelet-packets. En este caso, las señales codificadas de mayor calidad subjetiva son aquellas que tienen un alto contenido en ataques de señal. Por lo tanto, parece interesante diseñar un modelo de transitorios basado en MP donde el diccionario se construya a partir de funciones wavelet-packets.

Los átomos del diccionario para este modelo de transitorios se derivarán de un árbol wavelet-packets ortonormal. Se considerarán como átomos las respuestas de los filtros de síntesis de una descomposición WP completa,de forma que la correlación entre átomos y señal se obtenga simplemente filtrando la señal con el banco de filtros de análisis. De esta forma, el diccionario sobre-completo D_{WP} incluye todos los átomos de la descomposición wavelet-packets (WP) hasta un nivel de profundidad P, haciendo que el tamaño del diccionario sea de $M_{WP} = P \cdot N$, donde N es la longitud de la trama actual. Los átomos definidos de esta forma se pueden identificar a partir de los parámetros $\{s, p, r\}$, que indican la sub-banda s en la profundidad de descomposición p y el retardo r en la sub-banda s. El retardo real de la señal es proporcional a r y a la profundidad p, de la forma,

$$g_{\{s,p,r\}}[n] = g_{\{s,p\}}[n - 2^p r] \tag{6.4}$$

por lo que es fácil de determinar la posición de los transitorios modelados con un diccionario WP. Para obtener en el tiempo cada átomo $g_{\{s,p\}}[n]$, es necesario recorrer el árbol de síntesis de un banco de filtros wavelet-packets. Siendo $G_{\{s,p\}}(z)$ la transformada z de cada átomo, ésta se construye directamente a partir de los filtros $G_0(z)$ y $G_1(z)$, que se corresponden con las funciones de transferencia del filtro paso bajo y paso alto de los filtros de síntesis, respectivamente. El banco de filtros de síntesis se implementa, como indica la figura 6.8 (para el caso de $P = 3$), mediante la inserción de bloques de interpolación. Como resultado, cada átomo $G_{\{s,p\}}(z)$ se puede expresar como

$$G_{\{s,p\}}(z) = \prod_{d=0}^{p-1} G_{((\lfloor s/2^d \rfloor))_2}(z^{2^d}) \tag{6.5}$$

donde $((l))_L$ denota (l modulo L).

Para implementar el método *matching pursuits* es necesario, para la primera iteración, el calculo de la correlación entre la señal y los átomos. Para el caso del diccionario WP, los coeficientes wavelet-packets obtenidos mediante la transformada directa representan el peso de cada átomo:

$$\alpha^1_{\{s,p,r\}} = \langle x[n], g_{\{s,p,r\}}[n] \rangle \tag{6.6}$$

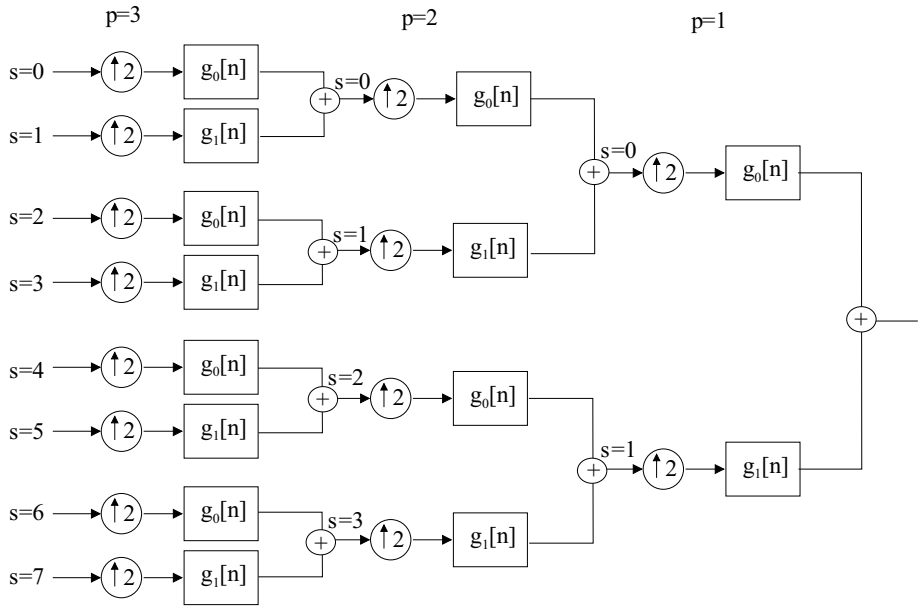

Figura 6.8: *Estructura en árbol de la transformada WP inversa con una profundidad de* $P = 3$.

Para el resto de iteraciones, las únicas correlaciones a calcular para implementar el método MP se corresponden con las correlaciones cruzadas entre todos los átomos y el átomo elegido en cada iteración. Por lo tanto, el problema de actualizar las correlaciones se puede reducir al cálculo de las correlaciones cruzadas entre átomos WP, $\langle g_{\{s_1,p_1,r_1\}}[n], g_{\{s_2,p_2,r_2\}}[n]\rangle$. Estas correlaciones se deben pre-calcular y tener almacenadas en memoria para realizar el proceso de actualización de las correlaciones inherente al método MP. Es muy importante expresar estas correlaciones convenientemente con el objetivo conocer la cantidad de memoria necesaria para guardar estos valores. En principio, guardar las correlaciones cruzadas entre todos los átomos puede requerir una memoria ingente, por lo que expresar estas correlaciones convenientemente ayuda a reducir la memoria necesaria para la implementación de *matching pursuits*. Además, si se conocen cuántas de estas correlaciones cruzadas son cero, se puede saber el número de multiplicaciones necesarias para actualizar las correlaciones (las correlaciones cruzadas que son cero no hay que actualizarlas). Por tanto, el primer paso consiste en expresar las correlaciones independientemente del retardo de cada átomo:

$$\begin{aligned} \langle g_{\{s_1,p_1,r_1\}}[n], g_{\{s_2,p_2,r_2\}}[n]\rangle &= \langle g_{\{s_1,p_1\}}[n-2^{p_1}r_1], g_{\{s_2,p_2\}}[n-2^{p_2}r_2]\rangle \\ &= \langle g_{\{s_1,p_1\}}[n-2^{p_1}r_1+2^{p_2}r_2], g_{\{s_2,p_2\}}[n]\rangle \end{aligned} \tag{6.7}$$

donde se supone $p_1 \geq p_2$ sin pérdida de generalidad. Con esta expresión, se logra reducir el número de correlaciones cruzadas a calcular (de las correlaciones entre todos los átomos) a sólo las correlaciones entre átomos con diferente sub-banda s y profundidad p, al eliminar el retardo r de la definición.

El proceso de simplificación de las correlaciones cruzadas no es sencillo, por lo que se tratará con detalle, pero, en cambio, el resultado es claro y contundente. Teniendo en cuenta que sólo las correlaciones entre átomos con relación de herencia ($s_2 = \lfloor \frac{s_1}{2^{p_1-p_2}} \rfloor$) son diferentes de cero, cuando los átomos se construyen a partir de una transformada ortonormal, la expresión (6.7) se

puede reducir a la siguiente [Vera04a]:

$$\langle g_{\{s_1,p_1,r_1\}}[n], g_{\{s_2,p_2,r_2\}}[n]\rangle = \begin{cases} \delta[r_2 - r_1] & s_1 = s_2, p_1 = p_2 \\ 0 & s_2 \neq \lfloor \frac{s_1}{2^{p_1-p_2}} \rfloor \\ g_{\{s,p,r_1\}}[r_2] & s_2 = \lfloor \frac{s_1}{2^{p_1-p_2}} \rfloor \end{cases} \tag{6.8}$$

donde $p = p_1 - p_2$ y $s = ((s_1))_{2^p}$. Así, de acuerdo con (6.8), la actualización de correlaciones sólo requiere guardar en memoria las respuestas al impulso de las ramas del árbol de síntesis WP hasta la profundidad $P-1$ (ver figura 6.8). Para realizar este proceso, basta con comprobar la relación entre el átomo a actualizar y el átomo elegido en cada iteración del MP. Cabe distinguir tres opciones:

1. Ambos átomos pertenecen a la misma profundidad y sub-banda. En ese caso, la correlación cruzada es cero, salvo que se trate del mismo átomo, que lógicamente es uno.

2. Ambos átomos no tienen relación de herencia. Entonces, su correlación cruzada es cero.

3. Ambos átomos tienen relación de herencia. En tal caso, la correlación cruzada es igual a la respuesta de la rama del árbol que hay que recorrer entre ambos átomos.

La complejidad depende del tamaño del diccionario, pero también del átomo que se elija en cada iteración del método MP, puesto que dependiendo de la profundidad del átomo seleccionado el número de correlaciones que son cero es diferente. El caso peor ocurre cuando se selecciona un átomo de profundidad $p = 1$, puesto que entonces el número de átomos correlados es máximo. En tal caso, hay $\frac{(P-1)\cdot N}{2}$ átomos con relación de herencia, lo que conlleva $\frac{(P-1)\cdot N}{2} = \frac{M_{WP}-N}{2}$ multiplicaciones reales y restas para realizar la actualización de correlaciones, según la ecuación (4.21) del método *matching pursuits*.

6.2.1. Demostración de las correlaciones cruzadas

Partiendo de la ecuación (6.7), la correlación que representa se va a escribir de la forma $c_{\{s_1,p_1\},\{s_2,p_2\}}[2^{p_2}r_2 - 2^{p_1}r_1]$. Esta correlación es igual a la correlación $c_{\{s_1,p_1\},\{s_2,p_2\}}[r_2]$ desplazada en el tiempo $2^{p_1}r_1$ muestras y diezmada por 2^{p_2}. En general, cualquier correlación se puede expresar por medio de la convolución mediante la expresión:

$$\langle g_{\{s_1,p_1\}}[n + r_2], g_{\{s_2,p_2\}}[n]\rangle = g_{\{s_1,p_1\}}[r_2] * g_{\{s_2,p_2\}}[-r_2] \tag{6.9}$$

La transformada z de la expresión (6.9) es simplemente,

$$C_{\{s_1,p_1\},\{s_2,p_2\}}(z) = G_{\{s_1,p_1\}}(z)G_{\{s_2,p_2\}}(z^{-1}) \tag{6.10}$$

Usando la expresión (6.5), suponiendo que $p_1 \geq p_2$, y teniendo en cuenta sólo los átomos con relación de herencia ($s_2 = \lfloor \frac{s_1}{2^{p_1-p_2}} \rfloor$), la ecuación (6.10) se puede expresar de la forma:

$$C_{\{s_1,p_1\},\{s_2,p_2\}}(z) = \Big(G_{\{s_2,p_2\}}(z) \prod_{d_1=p_2}^{p_1-1} G_{((\lfloor s_1/2^{d_1} \rfloor))_2}(z^{2^{d_1}})\Big) G_{\{s_2,p_2\}}(z^{-1}) \tag{6.11}$$

Sabiendo que $C_{\{s_2,p_2\},\{s_2,p_2\}}(z)$ es la autocorrelación de $g_{\{s_2,p_2\}}[r_2]$ en el dominio Z, la expresión (6.11) puede escribirse como:

$$C_{\{s_1,p_1\},\{s_2,p_2\}}(z) = C_{\{s_2,p_2\},\{s_2,p_2\}}(z) \cdot \prod_{d_1=p_2}^{p_1-1} G_{((\lfloor s_1/2^{d_1} \rfloor))_2}(z^{2^{d_1}}) \tag{6.12}$$

Ahora es necesario aplicar los cambios de variable $d = d_1 - p_2$ y $p = p_1 - p_2$. Considerando $s = ((s_1))_{2^p}$, la ecuación (6.12) se puede reescribir como:

$$C_{\{s_1,p_1\},\{s_2,p_2\}}(z) = C_{\{s_2,p_2\},\{s_2,p_2\}}(z) \cdot \prod_{d=0}^{p-1} G_{((\lfloor s/2^{d} \rfloor))_2}(z^{2^{d+p_2}}) = C_{\{s_2,p_2\},\{s_2,p_2\}}(z) \cdot G_{\{s,p\}}(z^{2^{p_2}}) \tag{6.13}$$

Volviendo ahora al dominio del tiempo, la ecuación (6.13) queda:

$$c_{\{s_1,p_1\},\{s_2,p_2\}}[r_2] = c_{\{s_2,p_2\},\{s_2,p_2\}}[r_2] * \left\{ \begin{matrix} g_{\{s,p\}}[\frac{r_2}{2^{p_2}}], \frac{r_2}{2^{p_2}} \in \mathbb{Z} \\ 0, \qquad otherwise \end{matrix} \right\} \tag{6.14}$$

Es preciso tener en cuenta que la autocorrelación $c_{\{s_2,p_2\},\{s_2,p_2\}}[r_2]$ es cero cuando $\frac{r_2}{2^{p_2}} \in \mathbb{Z}$, porque $g_{\{s_2,p_2\}}[r_2]$ es ortogonal a traslaciones de $2^{p_2} r_2$ muestras. Por lo tanto, al diezmar la expresión (6.14) por 2^{p_2} muestras, el resultado es $g_{\{s,p\}}[r_2]$. Finalmente, la correlación que se expresaba en la ecuación (6.7) para átomos con relación de herencia se simplifica a:

$$c_{\{s_1,p_1\},\{s_2,p_2\}}[2^{p_2} r_2 - 2^{p_1} r_1] = g_{\{s,p\}}[r_2 - 2^{p} r_1] = g_{\{s,p,r_1\}}[r_2] \tag{6.15}$$

Como se quería demostrar.

6.2.2. Resultados comparativos entre los diccionarios WP y EDS

Una vez se ha explicado cómo implementar el método *matching pursuits* con un diccionario WP y su complejidad asociada, el siguiente paso consiste en verificar su validez para el modelado de transitorios de señales de audio. En los experimentos que se han desarrollado para llevar a cabo esta tarea, se ha utilizado una trama de audio transitoria con un impulso de energía localizado en la mitad de la trama. Este impulso pertenece a un golpe de castañuela procedente de la señal de castañuelas (calidad CD mono), que es una de las señales propuestas por el grupo MPEG para su uso en actividades de estandarización [MPEG01]. Se ha implementado el modelo de transitorios con el diccionario wavelet-packets (WP) y con el diccionario de exponenciales amortiguadas (EDS) y comparado los resultados obtenidos.

En la figura 6.9, se presentan los resultados del modelado con el método *matching pursuits* y un diccionario EDS en iteraciones sucesivas. Como se observa en la figura, en las primeras iteraciones se extraen los transitorios fuertes de la señal, puesto que el diccionario está formado por señales que comienzan de esta forma. En las siguientes iteraciones, estos transitorios son suavizados para obtener la forma de la señal, lo que requiere un gran conjunto de átomos. Como consecuencia, se escogen muchos átomos de pequeña amplitud para refinar los errores cometidos por los primeros átomos. El diccionario EDS se define a partir de los parámetros: factor de amortiguamiento, frecuencia y retardo. La simulación se ha llevado a cabo con 10 factores de

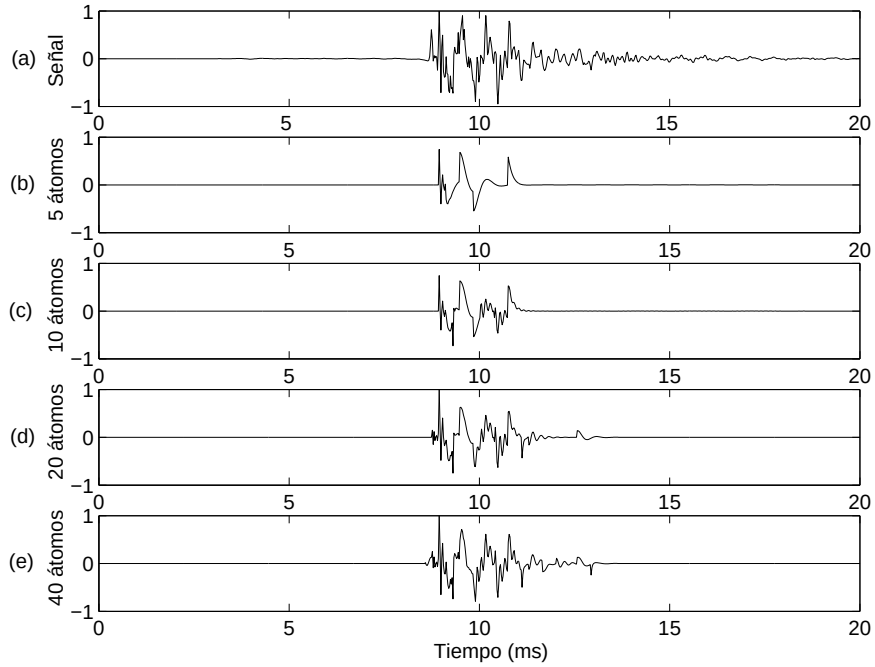

Figura 6.9: *La señal transitoria de un golpe de castañuela se modela mediante* matching pursuits *con un diccionario EDS. La señal original se presenta en (a) y la señal aproximada por este modelo en sucesivas iteraciones en: (b) 5 átomos, (c) 10 átomos, (d) 20 átomos y (e) 40 átomos.*

amortiguamiento, 32 frecuencias y $\frac{N}{8}$ posiciones de retardo, todos muestreados de forma lineal [Goodwin98]. Con estos valores, el tamaño del diccionario es de $M_{EDS} = 40 \cdot N$, donde N es el tamaño de la trama de análisis. La complejidad del método MP con un diccionario de exponenciales amortiguadas depende de la forma en que se actualicen las correlaciones. Aunque en la definición de MP [Mallat93] se proponga conocer todas las correlaciones cruzadas, según [Goodwin98] un notable ahorro en complejidad es posible si se implementa mediante un banco de filtros recursivos de un sólo polo. El número de multiplicaciones reales por iteración con este último método [Goodwin98] es de alrededor de $6 \cdot M_{EDS}$, donde M_{EDS} es el tamaño del diccionario EDS. En nuestro caso, el número de multiplicaciones por iteración queda $240N$, debido al carácter complejo del diccionario EDS definido.

El modelado con un diccionario WP se representa en la figura 6.10. Ahora, a diferencia del diccionario EDS, las transiciones fuertes no se extraen en las primeras iteraciones, y pese a ello no se observa la aparición de distorsión de pre-eco por efecto del modelado. Además, el transitorio se modela con mayor exactitud a partir de un número menor de átomos. Esto se debe, principalmente, a que no son necesarios átomos para compensar los errores iniciales. En este experimento, se ha implementado una descomposición WP con filtros de Daubechies de 32 coeficientes y profundidad de descomposición $P = 4$. Con este valor, el tamaño del diccionario es de $M_{WP} = 4 \cdot N$, y el número máximo de multiplicaciones reales máximo por iteración es de $\frac{3}{2}N$.

Parece lógico comparar el modelado obtenido con ambos diccionarios, en el sentido de comprobar cuál de ellos converge más rápidamente. Para este propósito se incluye la figura 6.11,

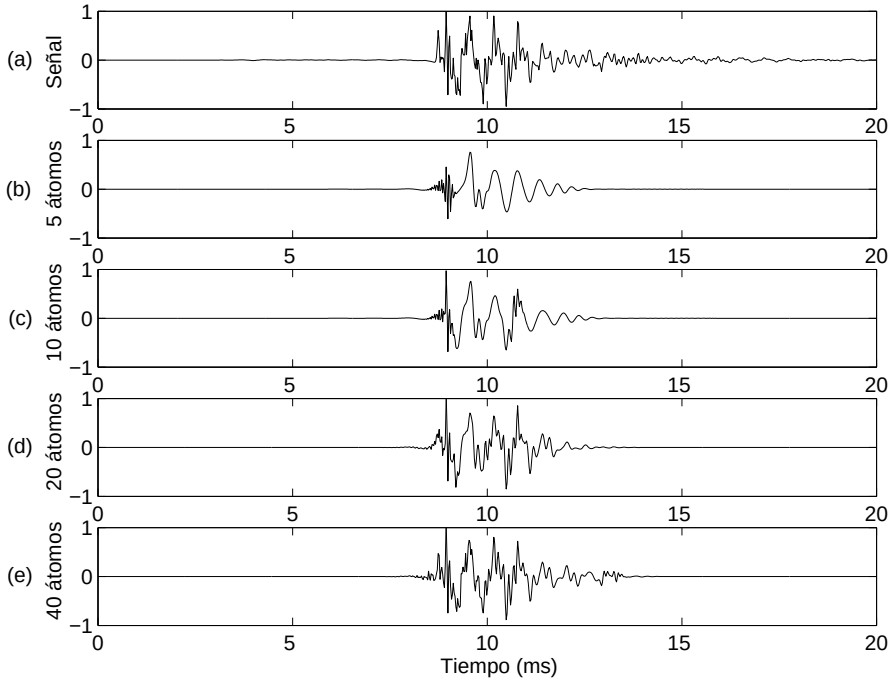

Figura 6.10: *La señal transitoria de una castañuela se modela mediante* matching pursuits *con un diccionario WP. La señal original se presenta en (a) y la señal aproximada por este modelo en sucesivas iteraciones en: (b) 5 átomos, (c) 10 átomos, (d) 20 átomos y (e) 40 átomos.*

donde se representa el error cuadrático medio (MSE, *Mean-Square-Error*) en dB entre la señal original y la aproximada por el modelo en función del número de iteraciones. Queda claro, a partir del resultado mostrado en la figura, que el diccionario WP converge más rápidamente que el EDS para el transitorio de audio utilizado. Esta diferencia es palpable, incluso, en las primeras iteraciones del *matching pursuits*, creciendo con el número de iteraciones. Por lo tanto, el número de átomos para un valor dado de error cuadrático medio es siempre menor cuando se utiliza un diccionario WP. Este resultado se consigue aunque el tamaño del diccionario WP sea 10 veces menor que el diccionario EDS usado, ya que se cumple $M_{EDS} = 10 \cdot M_{WP}$. Además, la complejidad (calculada como número de multiplicaciones reales por iteración) es menor en el caso del diccionario basado en funciones wavelet-packets.

Como conclusión, cabe decir que la aplicación del método *matching pursuits* con un diccionario basado en funciones wavelets-packets es una alternativa a tener en cuenta para implementar un modelo de transitorios para señales de audio. De inicio, la complejidad es baja al poder usar un tamaño de diccionario reducido y tener que calcular un número de multiplicaciones por iteración realmente pequeño. Finalmente, como se observa en los resultados, los transitorios sintetizados tienen una buena localización temporal, no existiendo distorsión de pre-eco debida al modelo.

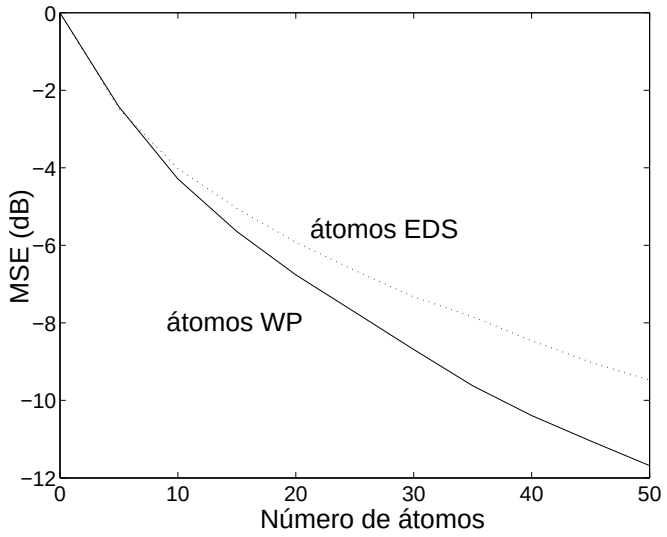

Figura 6.11: *Error cuadrático medio (MSE) de los modelos presentados en las figuras 6.9 y 6.10.*

6.3. Diccionario mixto: exponenciales complejas + wavelets packets

Los transitorios presentes en las señales de audio, como se puede observar en los diferentes ejemplos presentados, tienen una naturaleza muy diversa. Si el golpe de señal es muy fuerte, como en el caso de las castañuelas, la energía del transitorio predomina sobre el resto de la señal. Sin embargo, se pueden poner otros ejemplos, como es el caso del transitorio del gong, donde el incremento de energía no es tan brusco, existiendo en la misma señal una parte tonal y otra transitoria. Puede llegar el caso, como se verá posteriormente, que aparezcan lo que en la literatura conocen como micro-transitorios [Levine98]. En estos segmentos de señal predomina la parte tonal, siendo la energía de la parte transitoria apenas distinguible de la envolvente total de la señal. Así pues, como norma general, en los codificadores paramétricos de audio es imprescindible utilizar en cascada tanto el modelo tonal como el modelo de transitorios para separar ambas partes de la señal de entrada. Una iniciativa prometedora es el desarrollo de un diccionario mixto que permita extraer los parámetros de ambos modelos, obteniéndose así un modelo óptimo para la representación de señales de audio con transitorios.

En relación al orden de aplicación del modelo tonal o de transitorios, no hay en la literatura una postura común. Lo que sí es cierto es que el modelo de ruido se aplica sobre el residuo resultante de la aplicación de estos dos modelos en cascada. Así, en el codificador de Levine y Smith [Levine98], se utiliza la codificación por transformada para todo el segmento de señal considerado como transitorio, mientras que para el resto de segmentos se aplica en cascada un modelo tonal seguido de uno de ruido. En el codificador HILN [Purnhagen00], tras realizar un pre-análisis de la señal para detectar si hay transitorios, en el caso de que éstos existan se calcula la envolvente de la señal. La presencia del transitorio se indica al decodificador, en cuyo caso el tamaño del segmento de audio se reduce. Una vez hecho esto, se usa el modelo tonal, que en el caso de los transitorios se aplica sobre la señal dividida entre la envolvente estimada. En el codificador de Ali [Ali95], se aplican en cascada el modelo tonal, de transitorios y de ruido, y

por este orden. Tras el modelado tonal, se substrae la componente sinusoidal, dando lugar a un primer residuo que es la entrada al codificador de transitorios. En éste, la componente transitoria se extrae del primer residuo, generando un segundo residuo, que ya se codifica como ruido.

Sin embargo, al contrario que en el codificador de Ali, los más recientes codificadores paramétricos basados en un modelo de señal de tres componentes [Brinker02] [MPEG03] [Vera04c] [Verma00] (sinusoides, transitorios y ruido, STN) aplican primero el modelo de transitorios, seguido del modelo tonal y, por último, el modelo de ruido. La razón de este orden está en que el modelo tonal se adapta muy bien a la parte estacionario, de la señal de audio. La presencia de un transitorio cambia la estacionareidad local de la señal, empobreciendo el resultado que obtiene el modelo tonal. Por ejemplo, si el modelo tonal se usa para representar un transitorio fuerte de señal, el resultado será que el ataque se dispersará en el tiempo, dando como resultado una señal codificada con pre-eco. Extrayendo los transitorios de la señal antes de aplicar el modelo tonal, al menos se evita el problema del pre-eco. Con este enfoque de aplicación en serie de los modelos tonal y transitorio, respectivamente, se puede conseguir una buena calidad de la señal codificada para la mayoría de las señales de audio. Sin embargo, esta concatenación de modelos tiene dos inconvenientes principales:

- Como la aplicación de los modelos es en cascada (primero los transitorios y después los tonos), pueden aparecer problemas de separación entre componentes. En este sentido, pueden pasar dos cosas: por un lado, que el modelo de transitorios no extraiga todo el ataque pudiéndose producir un pre-eco final, y por otro, que los transitorios extraigan parte de la señal estacionaria, resultando un régimen binario total muy elevado.

- Cuando se aplica un segmento con un micro-transitorio al modelo de transitorios, la herramienta de modelado, a menudo, es incapaz de extraer este pequeño ataque, pasando directamente al modelo de ruido, dispersándose en el tiempo en la señal codificada.

A la vista de estos problemas, se propone modelar transitorios y tonos en la misma etapa del codificador. Para lograr esta herramienta, se utilizará el método *matching pursuits* con un diccionario mixto [Vera06a]. Este diccionario mixto debe incluir dos familias de funciones: 1) funciones que se adapten bien a las transiciones de señal; 2) funciones que puedan representar la parte estacionaria de la señal de audio. En este sentido, se han elegido funciones wavelet-packets para la parte transitoria y exponenciales complejas para la parte estacionaria. En una descomposición atómica de este tipo, en cada iteración se busca el átomo óptimo, que es el que extrae la mayor cantidad de energía del residuo actual. Dependiendo de las características del residuo en cada iteración, el átomo óptimo puede ser una exponencial compleja o una función wavelet-packets.

El principal problema del uso de diccionarios mixtos, como se dijo anteriormente, es la complejidad asociada a la implementación. En este sentido, debido a que los átomos son de diferente naturaleza, las correlaciones cruzadas entre todos los átomos, que deben estar almacenadas en memoria, suelen ocupar un tamaño prohibitivo para determinadas aplicaciones. A continuación, se presentará una implementación eficiente de un diccionario mixto formado por exponenciales complejas y funciones wavelet packets, basado en las propiedades particulares de ambas familias de funciones.

6.3.1. Planteamiento para una implementación rápida

El problema de la implementación del algoritmo *matching pursuits* con un diccionario mixto de exponenciales complejas y wavelet-packets radica en la actualización de las correlaciones cruzadas entre átomos de distinta naturaleza. Para realizar el resto de acciones como son: calcular la correlación entre la señal y los átomos, así como actualizar las correlaciones entre átomos de la misma naturaleza se pueden aprovechar los resultados obtenidos para los diccionarios individuales.

Por definición, el diccionario mixto $\mathcal{D}$ se forma mediante la unión de un diccionario de exponenciales complejas $\mathcal{D}_{EC}$ y un diccionario de funciones wavelet-packets $\mathcal{D}_{WP}$, quedando un diccionario formado por ambos tipos de familias de funciones $\mathcal{D} = \mathcal{D}_{EC} \cup \mathcal{D}_{WP}$. Los elementos del diccionario podrán ser exponenciales complejas $\mathbf{e}_k$, definidas por su frecuencia k, o wavelet-packets $\mathbf{w}_{\{s,p,r\}}$, definidas por la profundidad en la descomposición p, la sub-banda s en ese nivel de descomposición y el retardo r en la sub-banda actual.

En cada iteración del método MP, el algoritmo puede elegir como átomo óptimo una exponencial compleja o una función wavelet-packets. Se elegirá aquel átomo que extraiga del residuo actual la mayor cantidad de energía. Después, el procedimiento de actualización de correlaciones dependerá en gran medida del tipo de función que se haya extraído. Para explicar con detenimiento este procedimiento, se recordarán primero las propiedades de cada diccionario de forma individual.

Propiedades de las funciones wavelet-packets

Para el caso del diccionario wavelet-packets (WP) se va a restringir la familia de funciones wavelet-packets a una familia ortonormal, porque esto permite acelerar el proceso de actualización de correlaciones. El diccionario WP, $\mathcal{D}_{WP}$, se compone de todos los átomos del árbol de síntesis hasta una profundidad P, por lo que el tamaño del diccionario WP es $M_{WP} = P \cdot N$, siendo N la longitud de la trama de análisis. El cálculo de la correlación inicial para el método *matching pursuits* entre la señal y los átomos WP se limita a la obtención de todos los coeficientes de la descomposición WP hasta una profundidad P. El resultado serán los pesos $\boldsymbol{\alpha}^1_{\{s,p,r\}}$ en la primera iteración ($i = 1$), asociados a cada átomo WP mediante los tres índices: s sub-banda, p profundidad y r retardo.

Una vez calculadas las correlaciones iniciales, para realizar la actualización de correlaciones cuando se elige un átomo WP, basta con conocer las correlaciones entre todos los átomos y el átomo seleccionado. Cuando se quieren actualizar las correlaciones entre los átomos wavelet-packets es preciso determinar la relación: $\langle \mathbf{w}_{\{s_1,p_1,k_1\}}, \mathbf{w}_{\{s_2,p_2,k_2\}} \rangle$. Estas correlaciones se han presentado en la ecuación (6.8), y deben ser pre-calculadas y guardadas en memoria. Resumiendo, para actualizar las correlaciones entre átomos WP, basta con tener almacenado en memoria las respuestas impulsivas de cada rama del árbol de síntesis WP [Vera04a], y actualizar sólo los átomos con relación de herencia con el seleccionado (los que tienen correlación cruzada distinta de cero).

Propiedades de las exponenciales complejas

Para el diccionario de exponenciales complejas $\mathcal{D}_{EC}$, que extrae la parte estacionaria de la señal de audio, se propone el uso de un diccionario complejo sobre señales reales. De esta forma, sólo la frecuencia de cada exponencial forma parte de los índices de cada elemento del diccionario, reduciendo el tamaño del mismo (como se ha visto en el diccionario individual). Así pues, la información de fase se extrae directamente de las correlaciones (que son complejas). Con esta definición del diccionario de exponenciales complejas, cada función senoidal que se extrae de la señal es una combinación lineal de dos exponenciales complejas conjugadas.

En este caso, se supondrá, para simplificar las ecuaciones resultantes, que la ventana utilizada con el objetivo de limitar la duración del tamaño de trama a N muestras es una ventana rectangular. Con esta ventana, los átomos exponenciales complejos $\mathbf{e}_k$ se pueden escribir como:

$$e_k[n] = \frac{1}{\sqrt{N}} e^{j\frac{2\pi k}{2L}n}, \quad k = 0, \ldots, L \quad n = 0, \ldots, N-1 \tag{6.16}$$

donde k es la frecuencia discreta de cada átomo. La constante $\frac{1}{\sqrt{N}}$ se deriva del uso de la ventana rectangular para obtener átomos de energía unidad. El tamaño de diccionario para exponenciales complejas es de $M_{EC} = L+1$ átomos, y se corresponde con el número de frecuencias discretas.

Debido a la naturaleza compleja de estos átomos, la búsqueda del átomo óptimo, en la parte exponencial compleja del diccionario mixto, se realiza mediante subespacios conjugados, algo ya explicado en la implementación del modelo tonal. En la iteración inicial, el cálculo de los pesos $\boldsymbol{\alpha}_k^1$ asociados a cada frecuencia, es decir, la correlación entre la señal $\mathbf{x}$ y los átomos $\mathbf{e}_k \in \mathcal{D}_{EC}$, queda, para el caso de la ventana rectangular, simplemente como la DFT de la señal de entrada:

$$\langle \mathbf{x}, \mathbf{e}_k \rangle = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} x[n] e^{-j\frac{2\pi k}{2L}n} = \frac{1}{\sqrt{N}}\ X[k] \tag{6.17}$$

donde $X[k]$ es la DFT de longitud $2L$ de la señal de entrada $x[n]$, y el número de frecuencias debe cumplir $L > N$ para tener un diccionario sobre-completo de exponenciales complejas. Como $X[k]$ tiene valores complejos, contiene tanto la información de amplitud como la de fase. Estas correlaciones iniciales se pueden calcular mediante el algoritmo FFT, lo que implica rellenar con ceros la señal de entrada $x[n]$ hasta una longitud $2L$.

El procedimiento de actualización de correlaciones se complica al realizar la búsqueda en el subespacio conjugado. En general, se debe seguir la expresión (5.8) para su implementación. En el caso particular de actualizar correlaciones entre átomos exponenciales complejos, cuando un átomo de este tipo se elige como átomo óptimo, la expresión anterior se puede escribir como:

$$\langle \mathbf{r}^{i+1}, \mathbf{e}_k \rangle = \langle \mathbf{r}^i, \mathbf{e}_k \rangle - \alpha_{k(i)} \langle \mathbf{e}_{k(i)}, \mathbf{e}_k \rangle - \alpha^*_{k(i)} \langle \mathbf{e}^*_{k(i)}, \mathbf{e}_k \rangle \tag{6.18}$$

Como se demostró en el modelo tonal, cuando se trata de exponenciales complejas, estas correlaciones cruzadas se pueden calcular de forma eficiente mediante el algoritmo FFT. Así, para el caso particular de usar ventana rectangular, estas correlaciones se pueden expresar como:

$$\langle \mathbf{e}_{k(i)}, \mathbf{e}_k \rangle = \frac{1}{N} \sum_{n=0}^{N-1} e^{-j\frac{2\pi(k-k(i))}{2L}n} = \frac{1}{N} U[((k-k(i)))_{2L}] \tag{6.19}$$

$$\langle \mathbf{e}^*_{k(i)}, \mathbf{e}_k \rangle = \frac{1}{N} \sum_{n=0}^{N-1} e^{-j\frac{2\pi(k+k(i))}{2L}n} = \frac{1}{N} U[((k+k(i)))_{2L}] \tag{6.20}$$

donde ahora $U[k]$ es la DFT de longitud $2L$ de la función unidad $u[n]$ (como corresponde al cuadrado de la ventana rectangular). Esta DFT debe ser pre-calculada y guardada en memoria para una actualización rápida de las correlaciones en cada iteración del *matching pursuits*. Resumiendo, para el diccionario de exponenciales complejas, hay que calcular: 1) las correlaciones iniciales con la señal mediante una FFT de longitud $2L$; 2) las correlaciones cruzadas entre átomos, lo que requiere un vector de longitud $2L$ guardado en memoria.

En el caso de formar un diccionario mixto compuesto por un diccionario de exponenciales complejas y wavelet-packets ($\mathcal{D} = \mathcal{D}_{EC} \cup \mathcal{D}_{WP}$), el método *matching pursuits* debe calcular en cada iteración i los pesos $\{\boldsymbol{\alpha}^i_k, \boldsymbol{\alpha}^i_{\{s,p,r\}}\}$ asociados a los elementos del diccionario $\{\mathbf{e}_k, \mathbf{w}_{\{s,p,r\}}\}$. En la primera iteración, estos pesos son las correlaciones entre la señal de entrada y todos los átomos del diccionario. Los pesos en la iteración inicial se calculan mediante la Transformada Discreta de Fourier (DFT) y la Transformada Wavelet-Packets (WPT) de la señal de entrada, dependiendo de si se corresponden con las exponenciales complejas o con las funciones wavelet-packets, respectivamente. Una vez estos pesos son calculados, se elige el átomo óptimo (que es el que minimiza la energía del residuo), que puede ser o una exponencial compleja o una función wavelet-packets. A continuación, se debe llevar a cabo la actualización de las correlaciones, lo que implica conocer de antemano todas las correlaciones cruzadas entre los átomos del diccionario. Ya se han estudiado las correlaciones cruzadas entre los átomos de la misma familia. En este momento, se va a proceder a exponer cómo se obtienen las correlaciones cruzadas entre exponenciales complejas y wavelet-packets. Este cálculo de correlaciones entre átomos de diferente naturaleza depende del tipo de átomo extraído en cada iteración, lo que da lugar a dos situaciones diferentes que se estudiarán por separado.

6.3.2. Cálculo de la correlación cruzada entre una exponencial compleja elegida como átomo óptimo y funciones wavelet-packets.

En este caso, el átomo óptimo es complejo, mientras que la función extraída de la señal es real, ya que se trabaja con subespacios complejos. El procedimiento de actualización de estas correlaciones sigue la ecuación (5.8) del método *matching pursuits* con subespacios complejos. La implementación de esta ecuación requiere pre-calcular las correlaciones cruzadas entre el átomo óptimo $\mathbf{e}_{k(i)}$ en la iteración i y todas las funciones wavelet-packets $\mathbf{w}_{\{s,p,r\}} \in \mathcal{D}_{WP}$. Este cálculo se puede expresar en función de la DFT de las funciones wavelet-packets gracias a la naturaleza de las exponenciales complejas:

$$\langle e_{k(i)}[n], w_{\{s,p,r\}}[n] \rangle = \sum_{n=0}^{N-1} \frac{1}{\sqrt{N}} e^{j\frac{2\pi k(i)}{2L}n} w_{\{s,p,r\}}[n] = \frac{1}{\sqrt{N}} W^*_{\{s,p,r\}}[k(i)] \tag{6.21}$$

$$\langle e^*_{k(i)}[n], w_{\{s,p,r\}}[n] \rangle = \sum_{n=0}^{N-1} \frac{1}{\sqrt{N}} e^{-j\frac{2\pi k(i)}{2L}n} w_{\{s,p,r\}}[n] = \frac{1}{\sqrt{N}} W_{\{s,p,r\}}[k(i)] \tag{6.22}$$

donde $W_{\{s,p,r\}}[k(i)]$ es el valor de la DFT de longitud $2L$ de la función $w_{\{s,p,r\}}[n]$ en la frecuencia discreta $k(i)$. Notar que no es necesario guardar dos vectores, uno por cada ecuación, ya que, debido a la naturaleza real de las funciones wavelet-packets, el resultado de una es simplemente el conjugado de la otra. Por lo tanto, para poder realizar el procedimiento de actualización de estas correlaciones, hay que almacenar en memoria las transformadas DFTs de longitud $2L$ de todas las funciones wavelet-packets $w_{\{s,p,r\}}[n]$. Como el tamaño del diccionario WP, $\mathcal{D}_{WP}$, es $M_{WP} = N \cdot P$, el número de transformadas DFT de longitud $2L$ es de $N \cdot P$, lo que se considera un excesivo gasto de memoria.

Sin embargo, se puede ahorrar memoria teniendo en cuenta las propiedades de las funciones wavelet-packets. La idea es aprovechar la relación entre funciones cuando se varía el retardo r, pudiéndose escribir $w_{\{s,p,r\}}[n] = w_{\{s,p\}}[n - 2^p r]$. Con esta relación y, teniendo en cuenta las propiedades de desplazamiento en el tiempo de la DFT, se necesita guardar en memoria sólo las DFTs de longitud $2L$ de las funciones $w_{\{s,p\}}[n]$. En este caso, el número de DFTs de longitud $2L$ se reduce a $2^{P+1} - 2$, que es el número de nodos del árbol WP. Para comprobar este resultado, se aplican las propiedades de desplazamiento en el tiempo de la DFT:

$$\begin{array}{ll} \langle e_{k(i)}[n], w_{\{s,p\}}[n - 2^p r] \rangle & = \\ \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} e^{j\frac{2\pi k(i)}{2L} n} w_{\{s,p\}}[n - 2^p r] & = \\ \frac{1}{\sqrt{N}} e^{j\frac{2\pi k(i)}{2L} 2^p r} \sum_{l=0}^{N-1-2^p r} e^{j\frac{2\pi k(i)}{2L} l} w_{\{s,p\}}[l] & \end{array} \tag{6.23}$$

donde $l = n - 2^p r$, y se supone que las funciones wavelet-packets son causales, $w_{\{s,p\}}[l] = 0,\ \forall l < 0$. Esto obliga a realizar la transformada WP donde el modo de extensión sea el relleno con ceros. Se ha elegido este modo de extensión porque el modo periódico (desplazamientos circulares de las funciones con el retardo r) puede producir resultados inadecuados. Por ejemplo, si un transitorio está en los comienzos de una trama, su energía se puede dispersar al final de la misma debido a los desplazamientos circulares. No obstante, debido a las propiedades de la DFT para señales con desplazamientos circulares, se podría llegar a un resultado más compacto para actualizar las correlaciones con modo de extensión periódico.

Es posible relacionar el resultado de la ecuación (6.23) con la DFT de longitud $2L$ de cada función $w_{\{s,p\}}[n]$ de la forma:

$$\begin{array}{ll} \langle e_{k(i)}[n], w_{\{s,p\}}[n - 2^p r] \rangle & = \\ \frac{1}{\sqrt{N}} e^{j\frac{2\pi k(i)}{2L} 2^p r} (\sum_{l=0}^{N-1} e^{j\frac{2\pi k(i)}{2L} l} w_{\{s,p\}}[l] - \sum_{l=N-2^p r}^{N-1} e^{j\frac{2\pi k(i)}{2L} l} w_{\{s,p\}}[l]) & = \\ \frac{1}{\sqrt{N}} e^{j\frac{2\pi k(i)}{2L} 2^p r} (W^*_{\{s,p\}}[k(i)] - \sum_{l=N-2^p r}^{N-1} e^{j\frac{2\pi k(i)}{2L} l} w_{\{s,p\}}[l]) & \end{array} \tag{6.24}$$

donde $W_{\{s,p\}}[k(i)]$ es el valor de la DFT de longitud $2L$ de la función $w_{\{s,p\}}[n]$ en la frecuencia discreta $k(i)$. El sumatorio de la ecuación (6.24) se debe calcular para todos los posibles valores de r de cada función $w_{\{s,p\}}[n]$. Una posible forma de implementarlo es mediante un filtro digital con coeficientes complejos. Para reducir la complejidad asociada de este cálculo, este término se puede obtener para valores consecutivos de r. Con este enfoque, la complejidad asociada al sumatorio es de $N - 2^p$ multiplicaciones complejas para todos los valores de r de cada función $w_{\{s,p\}}[n]$. De forma adicional, la primera exponencial de la misma expresión representa una multiplicación compleja cada 2^p muestras. Resumiendo, el número de multiplicaciones para implementar la ecuación (6.24) es de $N - 2^p + \frac{N}{2^p}$ para cada función wavelet-packets $w_{\{s,p\}}[n]$. No se ha tenido en

cuenta, sin embargo, que las funciones $w_{\{s,p\}}[n]$ son funciones localizadas en el tiempo, de forma que sus amplitudes son cero a partir de un determinado valor en el tiempo. Esta propiedad implica una importante reducción del coste computacional. Para poder evaluar esta reducción, es necesario conocer la familia de funciones wavelet-packets usada para realizar la descomposición WP, siendo suficiente con saber la longitud del filtro paso bajo y paso alto asociada a cada etapa de la descomposición WP.

Como conclusión, cuando una exponencial compleja se elige como átomo óptimo, para actualizar las correlaciones con las funciones wavelet-packets es necesario: 1) guardar $2^{P+1}-2$ DFTs de longitud $2L$, una por cada función $w_{\{s,p\}}[n]$; 2) calcular el efecto del retardo ($w_{\{s,p,r\}}[n] = w_{\{s,p\}}[n-2^p r]$) para cada función $w_{\{s,p\}}[n]$, lo que corresponde a un máximo de $N-2^p+\frac{N}{2^p}$ multiplicaciones por función.

6.3.3. Cálculo de la correlación cruzada entre una función wavelet-packets elegida como átomo óptimo y exponenciales complejas.

Se trata ahora el caso en el que la función óptima en la iteración i pertenece a la familia de funciones wavelet-packets $\mathbf{w}_{\{s(i),p(i),r(i)\}}$, y se desean conocer las correlaciones cruzadas entre este átomo y todas las exponenciales complejas $\mathbf{e}_k \in \mathcal{D}_{EC}$:

$$\langle w_{\{s(i),p(i),r(i)\}}[n], e_k[n]\rangle = \frac{1}{\sqrt{N}}\sum_{n=0}^{N-1} w_{\{s(i),p(i),r(i)\}}[n]e^{-j\frac{2\pi k}{2L}n} = \frac{1}{\sqrt{N}}W_{\{s(i),p(i),r(i)\}}[k] \qquad (6.25)$$

donde $W_{\{s(i),p(i),r(i)\}}[k]$ es la DFT de longitud $2L$ de la función wavelet-packets $w_{\{s(i),p(i),r(i)\}}[n]$. De nuevo, para implementar el procedimiento de actualización de las correlaciones cruzadas en *matching pursuits* con el diccionario mixto, hay que guardar las DFTs de longitud $2L$ de cada función wavelet-packets. Pero, como antes, los requerimientos de memoria se pueden relajar aplicando la relación $w_{\{s(i),p(i),r(i)\}}[n] = w_{\{s(i),p(i)\}}[n-2^{p(i)}r(i)]$. En este caso, las correlaciones cruzadas quedan:

$$\begin{array}{ll} \langle w_{\{s(i),p(i)\}}[n-2^{p(i)}r(i)], e_k[n]\rangle & = \\ \frac{1}{\sqrt{N}}\cdot\sum_{n=0}^{N-1} w_{\{s(i),p(i)\}}[n-2^{p(i)}r(i)]e^{-j\frac{2\pi k}{2L}n} & = \\ \frac{1}{\sqrt{N}}e^{-j\frac{2\pi k}{2L}2^{p(i)}r(i)}\sum_{l=0}^{N-1-2^{p(i)}r(i)} w_{\{s(i),p(i)\}(m)}[l]e^{-j\frac{2\pi k}{2L}l} & = \\ \frac{1}{\sqrt{N}}e^{-j\frac{2\pi k}{2L}2^{p(i)}r(i)}(W_{\{s(i),p(i)\}}[k]-\sum_{l=N-2^{p(i)}r(i)}^{N-1} w_{\{s(i),p(i)\}}[l]e^{-j\frac{2\pi k}{2L}l}) & \end{array} \qquad (6.26)$$

donde $W_{\{s(i),p(i)\}}[k]$ es la DFT de longitud $2L$ de la función $w_{\{s(i),p(i)\}}[n]$, $l = n-2^{p(i)}r(i)$ y se ha considerado que el modo de extensión es relleno con ceros: $w_{\{s,p\}}[l] = 0,\ \forall l < 0$. El problema de obtener el último sumatorio en la ecuación (6.26) es ahora diferente. Es necesario calcular este valor para todas las posibles frecuencias discretas, $k = 0, \cdots, L$, con el valor de retardo $r(i)$ del átomo wavelet packets óptimo elegido en la iteración i. En función del valor concreto del retardo $r(i)$, el cálculo del sumatorio puede ser menos complejo mediante una FFT o un filtro digital con coeficientes complejos. En la práctica, la mayoría de las funciones $w_{\{s(i),p(i),r(i)\}}[n]$ tienen valores iguales a cero en las muestras desde $N-2^{p(i)}r(i)$ hasta $N-1$, lo que supone que el sumatorio no es necesario calcularlo.

Concluyendo, cuando se elige un función wavelet packets como átomo óptimo, el procedimiento de actualización de las correlaciones para las exponenciales complejas requiere: 1) guardar $2^{P+1}-2$ DFTs de longitud $2L$, una por cada función wavelet packets $w_{\{s,p\}}[n]$, y 2) calcular el efecto del retardo efectivo $2^{p(i)}r(i)$ del átomo óptimo, $(w_{\{s(i),p(i),r(i)\}}[n] = w_{\{s(i),p(i)\}}[n-2^{p(i)}r(i)])$, lo que conlleva como máximo $2L \cdot log_2 2L$ multiplicaciones (el cálculo de la FFT en el sumatorio de la ecuación (6.26)).

6.3.4. Resumen de la complejidad asociada

Para terminar, cuando se implementa *matching pursuits* con un diccionario mixto compuesto de exponenciales complejas y wavelet packets ($\mathcal{D} = \mathcal{D}_{EC} \cup \mathcal{D}_{WP}$), los requerimientos de memoria para realizar la actualización de las correlaciones son:

1. Una DFT de longitud $2L$ para almacenar la transformada discreta de Fourier de la ventana rectangular con el objeto de poder actualizar las correlaciones cruzadas entre funciones exponenciales complejas $\mathbf{e}_k$.

2. La respuesta impulsiva de todas las ramas del árbol de síntesis WP, $\mathbf{w}_{\{s,p\}}$, en el caso de las funciones wavelet packets.

3. $2^{P+1}-2$ DFTs de longitud $2L$ para almacenar las transformadas discretas de Fourier de la respuesta de cada rama del árbol de síntesis WP $\mathbf{w}_{\{s,p\}}$ requeridas para actualizar las correlaciones cruzadas entre funciones de diferente naturaleza.

En la iteración inicial, hay que calcular la DFT de longitud $2L$ y la transformada WP hasta una profundidad P de la señal de entrada $x[n]$. Con esto se consigue inicializar las correlaciones entre la señal y los elementos del diccionario mixto. El número de multiplicaciones para el resto de iteraciones viene dado por el procedimiento de actualización de correlaciones. Se necesita una multiplicación por átomo para actualizar las correlaciones según la expresión (4.21), donde se multiplica el peso del átomo óptimo con las correlaciones cruzadas. Cuando se elige como átomo óptimo una exponencial compleja, la actualización de correlaciones se realiza ahora mediante la ecuación (5.8), lo que conlleva dos multiplicaciones por átomo para actualizar exponenciales complejas y una multiplicación por átomo para actualizar las funciones wavelet packets (realizando una simplificación por ser funciones reales). Sin embargo, esta complejidad se podría conseguir si se pre-calcularan y almacenaran todas las correlaciones cruzadas entre átomos, algo que exige una cantidad de memoria ingente.

Se propone reducir la cantidad de memoria, a costa de incrementar de forma adicional la complejidad para obtener las correlaciones cruzadas. Aprovechando las propiedades de exponenciales complejas y wavelet packets, se obtiene un incremento de complejidad, según se indica:

1. Cuando se elige como átomo óptimo una exponencial compleja $\mathbf{e}_k(i)$, para actualizar las correlaciones con:

   a) Las exponenciales complejas $\mathbf{e}_k$. No hay complejidad adicional.

   b) Las funciones wavelet packets $\mathbf{w}_{\{s,p,r\}}$. Se sigue la ecuación (6.24), por lo tanto, el número de multiplicaciones máximo para cada una de las $2^{P+1}-2$ funciones $\mathbf{w}_{\{s,p\}}$ es de $N - 2^p + \frac{N}{2^p}$ multiplicaciones.

2. Cuando se elige una función wavelet packets $\mathbf{w}_{\{s(i),p(i),r(i)\}}$ como átomo óptimo, para actualizar las correlaciones con:

   a) Las funciones wavelet packets $\mathbf{w}_{\{s,p(,r\}}$. No hay un incremento de complejidad.

   b) Las exponenciales complejas $\mathbf{e}_k$. Se computa la ecuación (6.26). Ahora, el número de multiplicaciones es muy dependiente del retardo $r(i)$ del átomo óptimo. En el peor caso, siendo el retardo igual a $r(i) = N - 2^{p(i)}$, la complejidad adicional se deriva del cálculo de una FFT de longitud $2L$, lo que conlleva $2L \cdot log_2 2L$ multiplicaciones.

6.3.5. Resultados en señales de audio con transitorios

En primer lugar, es preciso hacer una reflexión sobre las señales a utilizar en las pruebas experimentales. Estas señales deben contener un transitorio de audio, aunque se debe verificar el modelo propuesto tanto para transitorios importantes como para transitorios poco significativos (caso de un micro-transitorio). En cuanto a las herramientas a comparar, el método *matching pursuits* se va a utilizar bajo tres aproximaciones diferentes: (1) usando un diccionario mixto de exponenciales complejas y wavelet packets; (2) usando un diccionario de exponenciales complejas y, seguidamente en cascada, un diccionario de wavelet packets; (3) usando un diccionario de wavelet packets seguido, en cascada, de un diccionario de exponenciales complejas.

El primer ejemplo que se va a utilizar para ilustrar las ventajas del modelo propuesto es el de un transitorio fuerte de audio. En este caso, la señal es un golpe de castañuela, aunque menos fuerte que el de los ejemplos del apartado anterior. La idea es que la trama de audio contenga una parte tonal, para ver la interacción entre la parte tonal y transitoria en cada modelo de señal. En la figura 6.12 se muestra el modelado de este transitorio. Se observa cómo la mejor discriminación entre la parte tonal y la transitoria se obtiene con el primer enfoque, correspondiente al diccionario mixto. La segunda aproximación (aplicación en serie de diccionario de exponenciales complejas y wavelet packets) produce un pre-eco, así como un suavizado del transitorio extraído. Para el tercer enfoque (aplicación en serie de wavelet packets y exponenciales complejas) se aprecia como crece el número de átomos dedicados a modelar la parte transitoria.

El criterio de parada usado para todos los casos es el siguiente: se detiene el método *matching pursuits* cuando un átomo extrae del residuo actual menos del 2 % de la energía de este residuo. Se elige este valor para obtener un residuo con propiedades estocásticas, es decir, para tratar de eliminar todas las componentes tonales o transitorias y, de esta forma, evitar artefactos en el residuo modelado de forma sintética como ruido [Schijndel03].

El segundo ejemplo, presentado en la figura 6.13, muestra el modelado de un micro-transitorio procedente de la señal *glockenspiel* del conjunto de señales de MPEG. La estructura de esta figura es la misma que la de la figura anterior. En este caso, de nuevo, el diccionario mixto obtiene la mejor descomposición. El micro-transitorio modelado por el segundo enfoque (aplicación en serie de exponenciales complejas y wavelet packets) tiene menos riqueza que el modelado por el diccionario mixto. El tercer enfoque ni siquiera es capaz de representar el micro-transitorio, puesto que no se llega a extraer ningún átomo wavelet packets de la señal inicial.

En ambas figuras, 6.12 y 6.13, el resultado del diccionario mixto es mejor que el obtenido para los diccionarios en cascada. Parece lógico que, cuando una trama de audio contenga una parte tonal y otra transitoria, el diccionario mixto dé mejores resultados, ya que evita que cada

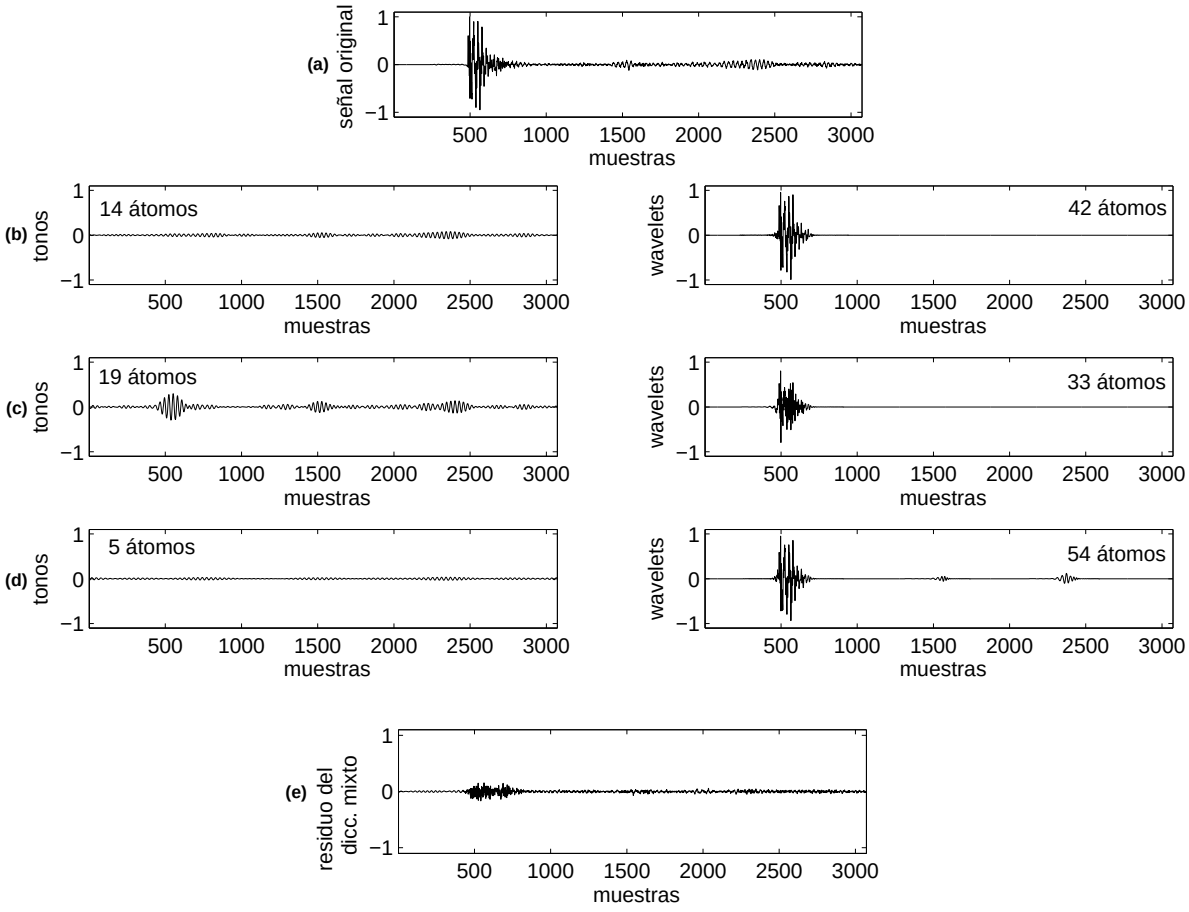

Figura 6.12: *(a) Trama de audio con un transitorio de castañuela. (b) Parte tonal y transitoria modelada con el diccionario mixto. (c) Parte tonal y transitoria modelada con un diccionario de exponenciales complejas seguido, en cascada, de un diccionario de wavelet packets. (d) Parte tonal y transitoria modelada con un diccionario de wavelet packets seguido, en cascada, de un diccionario de exponenciales complejas. (e) Residuo final del diccionario mixto.*

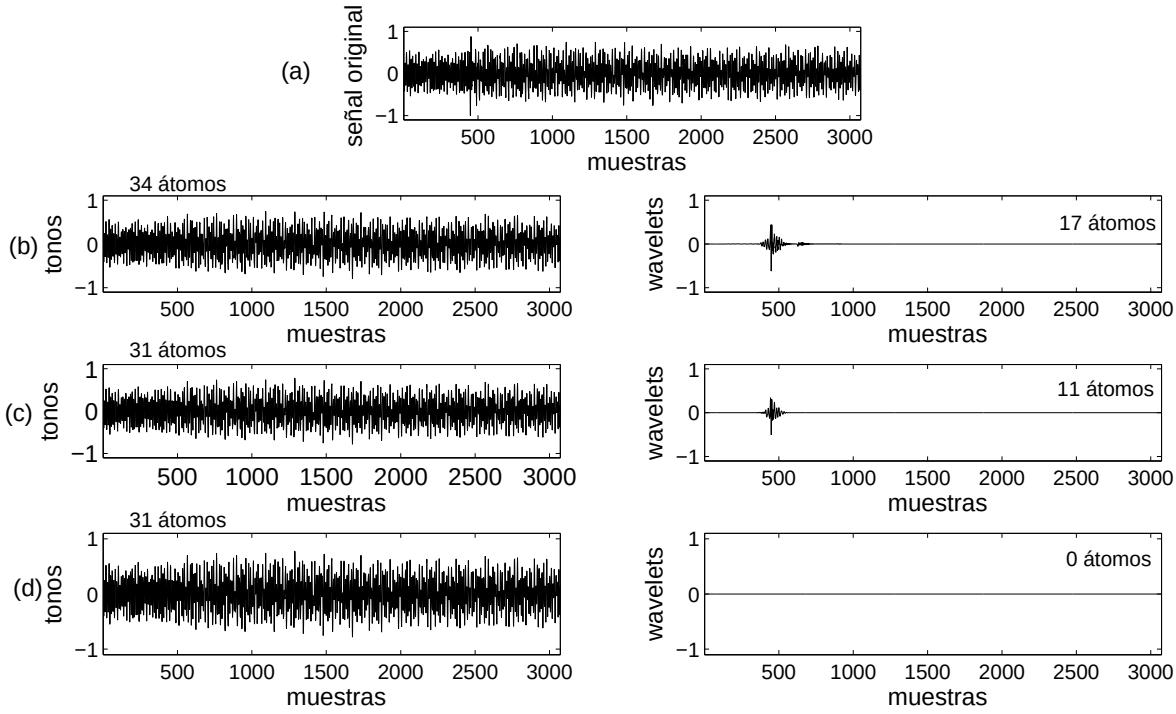

Figura 6.13: *(a) Trama de audio con un micro-transitorio. (b) Parte tonal y transitoria modelada con el diccionario mixto. (c) Parte tonal y transitoria modelada con un diccionario de exponenciales complejas seguido, en cascada, de un diccionario de wavelet packets. (d) Parte tonal y transitoria modelada con un diccionario de wavelet packets seguido, en cascada, de un diccionario de exponenciales complejas.*

Cuadro 6.1: *Preferencia de los resultados del diccionario mixto sobre el diccionario en serie en %.*

Fichero	Preferencia (%)
Suzanne Vega	55
Voz masculina en alemán	60
Voz femenina en inglés	70
Clavicordio	100
Castañuelas	100
Diapasón	52
Gaita	46
Glockenspiel	100
Punteos de guitarra	70
Sólo de trompeta	56
Pieza orquestal	60
Pop	100

diccionario simple cometa errores por exceso o por defecto en la extracción de átomos de la señal.

Para verificar que el uso del diccionario mixto también proporciona los mejores resultados perceptuales, se va a comparar con el uso en cascada del diccionario de exponenciales complejas y el diccionario wavelet packets (tercer enfoque). Se elige este esquema en serie porque es el más utilizado en los codificadores paramétricos que usan modelado de transitorios. Se han utilizado un subconjunto de señales de audio mono de calidad CD pertenecientes al grupo de señales recomendadas por MPEG [MPEG01] para tareas de estandarización. El esquema de análisis/síntesis implementado utiliza una ventana de Hanning con un tamaño de trama de $23ms$, es decir $N = 1024$ muestras, y un solapamiento del 50 % entre segmentos. El número de exponenciales complejas en $\mathcal{D}_{EC}$ es de $L + 1 = 4097$ átomos, mientras que la descomposición WP se ha llevado hasta una profundidad de $P = 4$, para un tamaño de $\mathbf{M}_{WP} = PN = 4096$ átomos, y se han usado filtros ortonormales de *Daubechies* con número máximo de momentos de anulación y 32 coeficientes. Notar que los resultados no cambian significativamente cambiando la profundidad de descomposición o la familia de los filtros wavelet packets [Vera04a]. Se han realizado unos tests de audición usando la metodología del triple estímulo con referencia ciega. Ahora, los resultados se muestran en la tabla 6.1.

Como se observa en la tabla 6.1, el diccionario mixto consigue los mejores resultados perceptuales para las señales con un alto contenido en transitorios. La explicación de este resultado es que el diccionario mixto evita, para los transitorios fuertes, artefactos de tipo *clicks*, mientras que para los micro-transitorios no produce distorsión de pre-eco (debida a la dispersión de la energía en el modelado de ruido si el micro-transitorio no ha sido extraído de la señal). Para las señales de audio muy tonales no hay diferencia perceptual entre los dos enfoques. Esta última reflexión es muy útil para reducir el coste computacional en un codificador paramétrico, puesto que el diccionario mixto tiene una complejidad alta y no está justificado su uso en tramas muy tonales de señal. Así pues, el método *matching pursuits* con un diccionario mixto es una herramienta ideal para modelar segmentos transitorios de audio que contengan una parte tonal aparte de la transitoria, ya que consigue un modelo preciso y bien localizado de la parte transitoria.

Capítulo 7

Modelado de ruido

Una vez obtenidos los parámetros de los tonos y de los transitorios, queda un residuo difícil de modelar. El problema principal se basa en que, a diferencia de tonos y transitorios tratados hasta ahora, el residuo no se puede modelar mediante una descomposición atómica. La causa se debe a que el residuo resultante de aplicar los modelos tonal y de transitorios es una señal poco correlada con los posibles átomos utilizados, por lo que no es posible implementar un modelo basado en descomposiciones atómicas que contenga en unos pocos átomos la energía de la señal residual. Muy al contrario, si se aplica cualquier descomposición atómica, el resultado será la dispersión de la energía en muchos átomos, cualquiera que sea el diccionario utilizado, ya que el residuo es el resultado de tratar la señal de audio con una gran variedad de átomos tiempo-frecuencia.

Sin embargo, si es posible implementar un modelo paramétrico para tratar el residuo. Este modelo parte de la falta de correlación de la señal con cualquier diccionario, por lo que la señal residual tiene características estocásticas, es decir, tiene propiedades similares a una fuente de ruido blanco filtrada. Así pues, el modelo que se utiliza normalmente para parametrizar la señal residual se conoce como modelo de ruido, puesto que realiza esta suposición sobre la señal residual.

En general, un modelo de ruido no parametriza la forma de onda de la señal, sino que simplemente extrae las envolventes de energía en frecuencia y tiempo como parámetros del modelo. En el decodificador, se genera ruido blanco que se filtra, a partir de la información de las envolventes, de forma que se obtenga la forma del residuo inicial en tiempo y frecuencia. A continuación, se describen los procedimientos de parametrización de ruido extraídos de la bibliografía, para hacer una comparación entre ellos y determinar los más validos para el propósito de esta tesis. Además, en este tema se tratarán otros aspectos relacionados con el modelo de ruido, como es la difícil separación entre tonos y ruido en un codificador paramétrico de audio.

7.1. El equilibrio imperfecto entre tonos y ruido

En los segmentos estacionarios de la señal de audio, la clave para obtener un modelo paramétrico que represente de forma satisfactoria la señal es la correcta separación entre tonos y ruido, es decir, la obtención de un equilibrio entre las partes determinística y estocástica de la señal. Idealmente, la parte sinusoidal abarca las componentes tonales del sonido, mientras que el ruido se ocupa de la parte estocástica. Por ejemplo, para el sonido de una flauta, la parte tonal es bien representada por tonos, mientras que el sonido que produce el aire se puede modelar como ruido.

Una separación incorrecta de las partes determinística y estocástica conduce a problemas tanto de calidad como de eficiencia en el modelo. Por un lado, si la parte sinusoidal no extrae todas las componentes tonales, éstas se modelarán como ruido. Como resultado, se producirán artefactos en el modelado, porque el modelo de ruido no es apto para generar señales tonales. Por otro lado, si el modelo tonal extrae más componentes que las propiamente tonales, se estará modelando ruido mediante tonos. La consecuencia será un régimen binario de codificación excesivamente elevado, así como la aparición de pitidos audibles en la señal proveniente del modelo tonal.

En la literatura aparecen un sinfín de enfoques para la solución de este problema [Thomson82] [Peeters98] [Levine99] aunque los dos más recientes [Schijndel03] [Myburg04] proponen soluciones encontradas. Es práctica habitual el empleo de información perceptual para la discriminación entre tonos y ruido, aunque las medidas de energía también son utilizadas. En [Schijndel03] aparece un mecanismo de fácil implementación para la separación entre tonos y ruido. Este enfoque se basa en dos herramientas principales:

- Primero, se aplica el modelo tonal sobre el segmento actual de la señal original. Esta extracción tonal se detiene en base a un criterio de parada perceptual. Este criterio se define para extraer en este paso todos los tonos perceptualmente importantes presentes en la señal, es decir, se detiene la extracción tonal cuando no quedan en la señal residual tonos perceptualmente significativos. Estos tonos son codificados para su envío al receptor.

- A continuación, sobre el residuo resultante del primer modelo tonal se aplica un segundo modelo tonal. En este caso, se extraen tonos hasta alcanzar un valor de compromiso basado en medidas de energía. Este valor de compromiso se define de forma que se siguen extrayendo tonos hasta que la energía extraída por un tono está por debajo de un umbral. La idea es extraer tonos hasta que la señal residual esté poco correlada con todos los tonos a diferente frecuencia. De esta forma, se supone que el residuo no tiene ya componentes tonales y está formado sólo por la parte estocástica de la señal. Los tonos extraídos por el segundo modelo tonal son descartados directamente, ya que son tonos enmascarados. El residuo resultante se pasa al modelo de ruido.

Aunque parezca que de esta forma se evitan todos los problemas comentados al principio de este apartado, ya que sólo se modelan tonos perceptualmente importantes y ruido de naturaleza estocástica, las imperfecciones del modelo paramétrico hacen que este enfoque no obtenga los mejores resultados en calidad perceptual al codificar la señal. Así, en [Myburg04] se cita que la señal de audio, aún en segmentos estacionarios, está formada por tonos, ruido y una mezcla de ambos. Teniendo en cuenta esta situación, la división del modelo entre tonos y ruido es demasiado simplista, haciendo difícil un equilibrio entre ambos. Según [Serra97], convertir el residuo en una señal estocástica simplifica enormemente su naturaleza, implicando que la componente determinística tenga que modelar todo lo que no sea estocástico. Se puede ver la distinción entre tonos y ruido como un proceso continuo, es decir, los tonos más importantes de la señal serán tonos puros, mientras que la señal también tendrá, en su caso, componentes de ruido estocástico. De un extremo a otro, los tonos se irán degradando, de forma que en el centro hay una región de ambigüedad donde se produce una transición entre tonos y ruido. Sin embargo, el modelo paramétrico asume una frontera clara entre ambas componentes, por lo que una separación

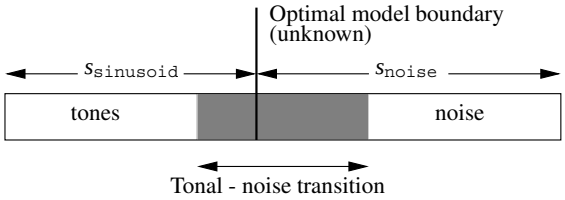

Figura 7.1: *La señal estacionaria de audio está formada por tonos, ruido y una mezcla de ambos. La frontera óptima entre tonos y ruido en un modelo de señal determinística más estocástica se desconoce a priori [Myburg04].*

perfecta es imposible de alcanzar.

La solución propuesta en [Myburg04] consiste en evitar que el modelo paramétrico descarte parte alguna de la señal de entrada. De esta forma, el codificador simplemente tendrá que decidir la frontera óptima de separación entre tonos y ruido, que es donde la señal modelada obtiene la mejor calidad perceptual a un reducido régimen binario. En la figura 7.1 se muestra gráficamente esta propuesta. Las causas de esta solución son varias, aunque la principal es que la eliminación de la parte central de la señal en la figura, que se propone en [Schijndel03], no conduce a la mejor calidad perceptual, ya que la señal que se descarta puede tener en su conjunto importancia perceptual. Además, la solución de la figura es muy útil en codificación escalable, puesto que si se mueve la frontera de separación a la izquierda se extraen menos tonos y el régimen binario disminuye progresivamente, así como la calidad perceptual.

Durante las pruebas realizadas en la definición del codificador paramétrico que se propone en esta tesis doctoral, se han implementado ambos enfoques. Los resultados dan la razón a la propuesta de [Myburg04] por los problemas que aparecen si se eliminan los tonos enmascarados del modelo paramétrico. Así, al implementar la solución propuesta en [Schijndel03], se encontraron los siguientes problemas:

- Se encuentran frecuentes pitidos en la señal codificada, principalmente en zonas con predominio de la componente ruidosa en la señal original. La causa de este efecto indeseado está en la extracción de tonos cuando la señal es de naturaleza ruidosa. Como se observa en la figura 5.16 del modelo sinusoidal, cuando el ruido es importante, se pueden extraer tonos donde hay ruido en la señal, incluso usando PMP como medida perceptual. Si, además, se elimina con el segundo modelo tonal gran parte de la energía de la banda donde se extrajo el tono erróneamente, se produce un pitido en la señal codificada.

- La señal se escucha filtrada y de alguna forma no natural en algunos momentos. Este es el efecto perceptual de descartar los tonos no audibles. Si los tonos no audibles eliminados se suman a la señal codificada final, este efecto perceptual desaparece.

Es preciso notar que los resultados son muy variables de unas señales a otras en función del umbral de energía escogido. No se ha podido encontrar un umbral válido para todas las señales de prueba, ya que el umbral que funciona de forma aceptable para unas señales no es apto para otras y viceversa.

Cuando se han realizado las pruebas con la propuesta de [Myburg04], el primer problema que ha surgido es de implementación. Como aparece en la figura 7.1, debe haber una frontera

ideal para la separación entre tonos y ruido. Esta frontera sólo se puede conocer a posteriori, es decir, cuando se evalúan los resultados entre varias señales codificadas con diferentes fronteras de separación. Como primera aproximación, se puede establecer la frontera en el punto en que todos los tonos perceptualmente importantes son modelados por el modelo tonal. Esto es lo mismo que poner la frontera (a la izquierda en la figura) en un punto donde el régimen binario final es bastante reducido. Si se extraen (y codifican) más tonos de los perceptualmente importantes el régimen binario crecerá y, es de esperar, que la calidad de la señal codificada mejore. Sin embargo, la calidad perceptual crecerá hasta el momento en el que se empiece a modelar en las zonas ruidosas de la señal el residuo con muchos tonos. En ese momento, se producirán pitidos en la señal codificada. En las pruebas realizadas, se ha usado como criterio extraer y codificar sólo los tonos perceptualmente significativos, de forma que se codifica como ruido el resto del residuo. Los artefactos más comunes encontrados mediante esta aproximación son:

- La señal codificada se escucha ruidosa en algunos segmentos. Concretamente, se escucha ruidosa cuando los tonos presentes en la señal no son tonos muy puros. Esto sucede en las señales vocales de naturaleza sonora, y en instrumentos musicales cuando se está cambiando entre una nota musical y otra (sobre todo en la trompeta). En estos casos, el modelo sinusoidal extrae pocos tonos, dejando un residuo con parte deteminística al modelo de ruido.

- Para algunas señales, como las vocales, se diferencian claramente como dos fuentes independientes la parte tonal y la ruidosa. Este efecto es muy común en segmentos sonoros de la señal vocal, puesto que en estos casos el residuo tiene una gran parte determinística. Cuando se modela como ruido este residuo, el oído interpreta la parte ruidosa como de naturaleza independiente a la vocal, pareciendo, de alguna manera, que se escucha la señal original filtrada más un eco ruidoso.

Estos artefactos, sin embargo, tienen menor importancia que los encontrados en el enfoque que elimina los tonos no audibles. Esta afirmación se realiza basándose en que estos defectos aparecen principalmente para señales vocales y que se pueden minimizar extrayendo (y codificando) más tonos (de los perceptualmente importantes) en el codificador. Sin embargo, esta estrategia no se va a seguir en la definición del codificador propuesto porque conduce a un régimen binario mayor y afecta (principalmente) a señales vocales. Para estas señales, ningún codificador paramétrico consigue mejor calidad que los vocoders más usuales. Como conclusión, los codificadores paramétricos, basados en la descomposición en un mismo segmento de señal estacionaria de tonos y ruido, se deben utilizar en aplicaciones donde la señal sea una señal musical. Como futura línea de investigación, se propone la variación del modelo de ruido para incluir en la generación del mismo, aparte de ruido blanco, la posibilidad de generar ruido multipulso [Ding97]. Con esta opción, se podrá modelar de forma más satisfactoria el residuo cuando éste posee una parte determinística proveniente de tonos no eliminados en la señal.

El difícil equilibrio entre tonos y ruido es aún más crítico que en el caso de segmentos transitorios de señal de audio. En este caso, los criterios perceptuales no son viables, ya que el umbral de enmascaramiento no es válido en segmentos no estacionarios de señal. Así pues, cuando se modela un transitorio, la frontera entre el modelado de transitorios (junto con tonos) y el modelado de ruido es, inevitablemente, una frontera basada en energía. En este caso, conforme más

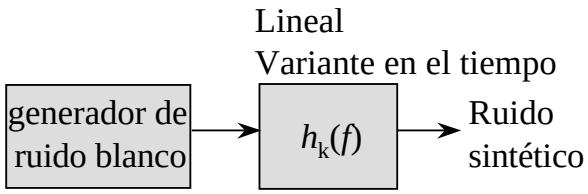

Figura 7.2: *Generador de ruido sintético.*

energía extraiga el modelo de transitorios, mayor será el régimen binario de la señal resultante y menor el pre-eco producido por la dispersión inherente al modelo de ruido. Esta dispersión se refiere a la falta de definición temporal del modelo de ruido. Aún cuando se incluyen como parámetros la envolvente temporal del residuo, esta envolvente, aunque introduce mejoras, es insuficiente para evitar pre-ecos en la señal codificada. En el codificador propuesto se utilizará un valor de compromiso que haga prácticamente inaudible el efecto de pre-eco. Esto es posible, incluso a bajo régimen binario, porque los segmentos transitorios son muy escasos en la señal de audio y el régimen binario que necesitan es, por lo tanto, sólo una pequeña parte del global.

7.2. Parámetros de la energía del residuo en frecuencia

En general, el modelo de ruido es un modelo de una fuente estocástica que se obtiene mediante filtrado. Por eso, la primera herramienta usada para el modelo de ruido fue la codificación mediante predicción lineal (LPC). En un modelo de este tipo, el codificador extrae mediante LPC los coeficientes de un filtro todo polos. Estos parámetros informan de la envolvente de la energía del residuo en frecuencia. En el decodificador se genera ruido blanco, que se filtra con los coeficientes del LPC obtenidos en codificación, resultando un ruido sintético con una envolvente en frecuencia similar al residuo original. Este modelo tan sencillo es, en realidad, un filtro variante en el tiempo que colorea una fuente de ruido blanco. Se dice que es variante en el tiempo porque los parámetros de la envolvente en frecuencia varían de una trama a otra de ruido. Con esta estructura, sin embargo, hay una gran cantidad de opciones de implementación, aunque en todas ellas los parámetros extraídos describen la forma variante en el tiempo de la energía en frecuencia del residuo. Así pues, el modelo del ruido en frecuencia se basa en un esquema similar al de la figura 7.2. Sin embargo, la forma de implementar este filtro variante en el tiempo es diferente en cada aproximación tomada de la bibliografía.

En general, dos estrategias radicalmente distintas se han impuesto en codificación paramétrica de audio. Por un lado, se tienen los bancos de filtros ERB, fácilmente implementables mediante la STFT, y por otro lado, las estrategias de *warped-LPC*, que son una simple modificación de los filtros LPC para obtener un espectro logarítmico en frecuencia. Es preciso tener en cuenta en el análisis del ruido en frecuencia que el residuo, aún teniendo en principio que estar compuesto sólo de la parte estocástica de la señal de audio, en realidad contiene una parte determinística, como sucede con los tonos no audibles en el caso del codificador propuesto en esta tesis. Por lo tanto, se explicarán detalladamente ambas estrategias y se implementarán para comprobar con cuál de ellas se obtiene una mejor calidad perceptual al sintetizar el ruido.

7.2.1. Bancos de filtros ERB

Un banco de filtros basado en criterios perceptuales fue introducido en [Goodwin96]. La idea de este banco de filtros es que el ancho de banda de cada filtro estuviera relacionado de alguna manera con las bandas críticas. Esto es así porque en algunas pruebas con señales ruidosas [Zwicker90] se ha estimado que el oído sólo es capaz de determinar la potencia de ruido en cada banda crítica. Esto quiere decir que si dos ruidos diferentes están confinados en frecuencia en una determinada banda crítica y tienen la misma potencia son indistinguibles. Este resultado se deriva de pruebas psicoacústicas con señales ruidosas de banda estrecha. Si un ruido de banda estrecha, con un ancho de banda menor que una banda crítica, aumenta su ancho de banda (manteniendo constante su potencia) no se nota diferencia alguna hasta que el ancho de banda en cuestión excede el de la banda crítica. A partir de estas pruebas, se deriva que la forma espectral del ruido no es tan crucial como que se mantenga la energía en cada banda crítica. Estas pruebas con ruido sirvieron para medir el ancho de banda de las bandas críticas en función de la frecuencia central del ruido. El resultado fue que el ancho de banda de cada banda crítica es de alrededor de 500 Hz para las bajas frecuencias, creciendo linealmente este valor con la frecuencia. Este resultado confirmaba la idea de que el filtrado que realiza el sistema auditivo es propio de un banco de filtros de Q constante (en medias y altas frecuencias). Resultados experimentales más actuales sugieren que el ancho de banda de las bandas críticas de baja frecuencia está relacionado de forma cuadrática con la frecuencia central [Moore83]. Las expresiones de estos anchos de banda de equivalencia rectangular (ERB, *Equivalent Rectangular Bandwidth*) para los filtros auditivos difieren, por tanto, algo de la teoría clásica de las bandas críticas, aunque en la práctica son muy similares. Realmente, no es fundamental realizar un banco de filtros variante en el tiempo para el ruido con una respuesta exacta al valor de los filtros auditivos, puesto que el ruido es una señal que por definición ocupa un gran ancho de banda y, por tanto, varios filtros del banco de filtros auditivos. Teniendo en cuenta la teoría de percepción del ruido, en [Goodwin96] se propone introducir un banco de filtros con anchos de banda ERB (*Equivalent Rectangular Bandwidth*).

Una explicación detallada de los bancos de filtros ERB se encuentra en [Goodwin97]. A continuación, sólo se realiza una breve exposición sobre ellos. Un banco de filtros ERB se ajusta a la idea de un filtro variante en el tiempo, aunque su implementación puede ser muy diversa. Una primera aproximación, tomada en [Goodwin97], es realizar un banco de filtros FIR cuyas respuestas mantengan el principio de ERB entre las bandas. En una estrategia de este tipo, en el codificador se hace pasar la señal residuo por este banco de filtros, obteniendo a su salida la energía en cada banda y en cada trama de ruido. En el decodificador basta con generar ruido blanco, filtrarlo con el banco de filtros ERB modificando la energía en cada banda por la energía enviada por el codificador e implementar los filtros de síntesis. Otro método de implementación de un banco de filtros ERB es mediante la STFT [Verma99]. La idea es utilizar las mismas herramientas matemáticas que usa el modelo sinusoidal. Aunque, a decir verdad, no se puede integrar en el decodificador ambos modelos, es decir, no se puede realizar sólo una transformada de Fourier inversa por trama porque ambos modelos tienen diferente tamaños de trama o número de muestras de las DFTs. En frecuencia, el modelo de ruido ERB es una herramienta de análisis/sintésis. En análisis, se estima en frecuencia la energía en cada banda ERB. Durante la síntesis, la energía de cada banda se controla fácilmente en frecuencia. El análisis basado en STFT calcula la energía dividiendo la transformada de cada trama en b bandas según el modelo

ERB. Para la trama t, la energía de cada banda se calcula de la forma:

$$E_b^t = \frac{1}{M} \sum_{k \in \beta_b} |X^t[k]|^2 \tag{7.1}$$

donde E_b^t es la energía de la banda ERB b en la trama t, β_b son las muestras de la DFT que pertenecen a la banda ERB b y $X^t[k]$ es la muestra k de la DFT de M muestras de la señal residual en la trama t. Estos parámetros de energía se pueden utilizar en el decodificador mediante un simple mecanismo de solapamiento (que suele haberlo entre tramas) y suma. Una nota importante es que el cambio del número de bandas ERB es muy sencillo en frecuencia, bastando con conocer las muestras que cada banda engloba. Sin embargo, esta modificación necesita cambiar el banco de filtros con un enfoque de filtros FIR. Se puede decir que mediante la técnica ERB el espectro de magnitud para sintetizar el ruido es un espectro lineal a trozos, ya que lo que hace el banco de filtros basado en STFT es modificar la varianza del ruido blanco (que lo es en parte real e imaginaria) a una varianza constante cada banda ERB. Aunque pueda parecer simplista esta aproximación del espectro, está basada en principios perceptuales y los buenos resultados obtenidos se emplean en algunos codificadores paramétricos de audio [Levine98] [Verma99].

7.2.2. Filtros basados en *warped-LPC*

El uso de coeficientes LPC que implementan un filtro IIR todo polos (P polos), mediante la predicción de una muestra a partir de P muestras de entrada, fue usado por primera vez en codificación de audio en [Serra89]. Realizando esta extracción de coeficientes en tiempo corto, es decir en una trama de análisis pequeña, se puede conseguir el filtrado variante en el tiempo que caracteriza el modelo de ruido. Para evitar problemas de inestabilidad en la cuantificación de los polos, se suelen utilizar los coeficientes de la estructura en celdas (*lattice*) [Markel76], verificando el test de estabilidad de Schür-Cohn. Además, se ha llegado a utilizar una excitación multipulso [Ding97] para conseguir mayor fidelidad cuando el residuo tiene una frecuencia fundamental (*pitch*) definida. Sin embargo, los coeficientes del filtro LPC se calculan minimizando la energía del error entre el ruido sintético generado y el residuo actual, de ahí el enfoque de predicción lineal. Esa minimización de la energía no está adaptada a la percepción del oído humano, ya que éste tiene una precisión logarítmica en la frecuencia. Además de este factor, la diferencia perceptual entre residuo y ruido sintético es función de la máscara actual. En relación al carácter logarítmico de la percepción en frecuencia, parece más lógico modelar con más detalle el espectro de baja frecuencia, donde hay muchas bandas críticas, que el de alta frecuencia. Con esta finalidad surgió la variación conocida como *warped-LPC* [Strube80]. En general, esta técnica sirve para incrementar la resolución del filtro todo polos en una cierta zona de frecuencia, a expensas de la resolución en el resto de la frecuencia. En este sentido, en lo que a codificación de audio se refiere, se pueden conseguir resoluciones en frecuencia similares a las obtenidas por algunas escalas con carácter logarítmico, como la escala de Bark o las bandas críticas, mediante una transformación bilineal de la transformada z [Smith95] de los filtros todo polos. Se han realizado en la bibliografía implementaciones a partir de estructuras de filtros conocidas [Harma00b] [Brinker03]. Este tipo de filtros basados en *warped-LPC* son los más utilizados en codificación paramétrica de audio, siendo su exponente más destacado el codificador estandarizado PPC (*Philips Parametric Coder*).

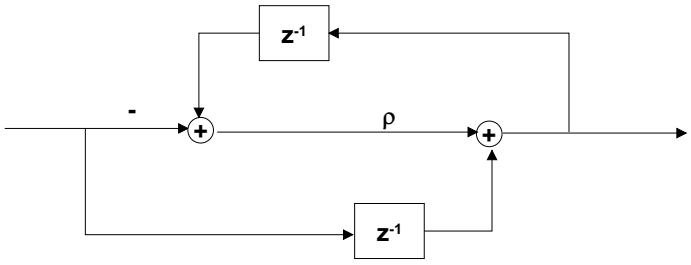

Figura 7.3: *Bloque a sustituir por cada retardo unidad para obtener filtros* warped.

La transformación de escala en frecuencia a banda de Bark es un problema superado [Smith99] [Harma00a]. Generalmente, un banco de filtros con anchos de banda uniformes en frecuencia se puede convertir en un banco de filtros con anchos de banda uniformes en banda de Bark (y por tanto logarítmico en frecuencia) mediante una transformación paso todo, usando una relación bilineal definida por la siguiente sustitución en el dominio z:

$$z = A_\rho(\zeta) \triangleq \frac{\zeta + \rho}{1 + \zeta\rho} \tag{7.2}$$

con la que se convierte el círculo unidad en el plano z en otro círculo unidad en el plano ζ tal que, para $0 < \rho < 1$, las bajas frecuencias se estiran y las altas se comprimen, en la misma forma que lo hace una transformación de frecuencia a escala de Bark. El valor del parámetro ρ depende de la frecuencia de muestreo de la señal original [Smith99]. Si se aplica la expresión (7.2) con $\rho = 0,756$ (y frecuencia de muestreo de entrada de $44,1kHz$) en un filtro, la escala obtenida es muy similar a la escala de Bark [Harma00a]. Para hacerse una idea del cambio de escala que se obtiene, la escala de Bark se puede aproximar a partir de las posiciones en frecuencia como [Zwicker99]:

$$b = 13arctan(0{,}76f(kHz)) + 3{,}5arctan(\frac{f(kHz)}{7{,}5})^2 \tag{7.3}$$

Un banco de filtros *warped* se puede obtener de forma inmediata a partir de un banco de filtros uniforme sustituyendo cada retardo unidad de la estructura del filtro por un bloque de primer orden paso todo que realiza la transformación bilineal de la expresión (7.2). Este bloque se obtiene escribiendo ζ en función de z de la forma [Harma00a]:

$$\zeta^{-1} = \frac{z^{-1} - \rho}{1 - \rho z^{-1}} \tag{7.4}$$

Según la expresión (7.4), el bloque se puede implementar mediante una sola multiplicación con la estructura que aparece en la figura 7.3. Este bloque realmente modifica la fase, y por tanto el retardo, para realizar el cambio de escala. Esta técnica, presentada en [Harma00b], permite implementar un banco de filtros *warped* sin cambiar los coeficientes ni la estructura del filtro origen, sólo se sustituyen los retardos unidad por el bloque paso todo. La inclusión de este bloque provoca un cambio de la fase con respecto al retardo unidad que permite procesar más lentamente las bajas frecuencias y más rápidamente las altas frecuencias. Para entender el funcionamiento de la transformación bilineal, es necesario poner un ejemplo. Los tres tonos de la figura 7.4 se introducen en una cadena de 1000 bloques de primer orden paso todo con $\rho = 0,723$.

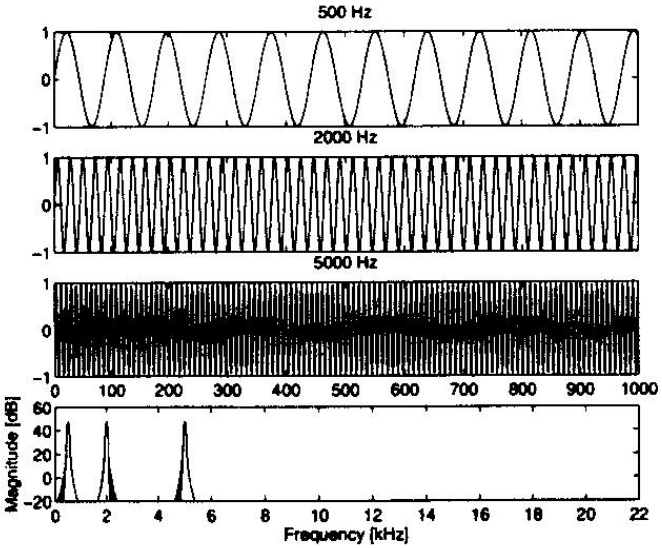

Figura 7.4: *Tres tonos en tiempo y frecuencia antes de realizar un procesado* warped *[Harma00a].*

La señal de salida después de procesar 1000 muestras se dibuja en la figura 7.5. Comparando ambas figuras, se observa como los tonos se propagan a diferente velocidad por la cadena de bloques paso todo, provocando así el cambio de escala esperado a la salida. Se puede interpretar este cambio de escala como el re-muestreo de la señal de salida en función de la frecuencia, que al fin y al cabo es la base del cambio de escala pretendido.

En cuanto a implementación, se puede realizar un enfoque basado en DFT modificada logarítmicamente o *warped*-DFT, como en el caso del banco de filtros ERB. Sin embargo, es más común emplear un enfoque de predicción lineal donde el codificador soluciona la matriz de autocorrelación, teniendo en cuenta en el sistema el cambio del retardo unidad por el bloque paso todo [Harma00a]. Este software está disponible como una librería de MATLAB llamada *warpTB* en la dirección: "*http://www.acoustics.hut.fi/software/warp/*". Una vez calculados los coeficientes del filtro, parece lógico convertirlos a los coeficientes de la estructura en celdas mediante el algoritmo de Schür, porque de esta forma se evitan inestabilidades en el proceso de cuantificación. Éstos valores son los enviados al receptor, que puede, bien usar la estructura en celdas, o bien obtener de nuevo los coeficientes de la estructura directa. Es interesante observar la forma de la envolvente del espectro obtenida por un modelo *warped*-LPC. En este sentido, en la figura 7.6 se representa la aproximación del espectro que realiza tanto un modelo lpc y otro *warped*-lpc con 40 polos. En primer lugar, se dibuja la frecuencia en escala lineal. En esta parte sólo se refleja que la versión *warped* tiene una mayor definición en bajas frecuencias. Sin embargo, la diferencia se aprecia en la segunda parte de la gráfica, donde se representa la frecuencia en escala logarítmica. Se observa ahora como la versión *warped* tiene una buena definición en bajas frecuencias a costa de la definición en altas frecuencias, de ahí que modele muy bien la energía en baja frecuencia. Es importante tener en cuenta que ésta es la escala empleada por el oído humano para la percepción de sonidos. Se puede admitir que para el sistema auditivo el error es más grave en función de las bandas críticas que ocupa (además de su amplitud), no a partir de la distancia en frecuencia lineal. Bajo esta premisa, en[Harma00a] se afirma que el modelo *warped*-lpc minimiza el error

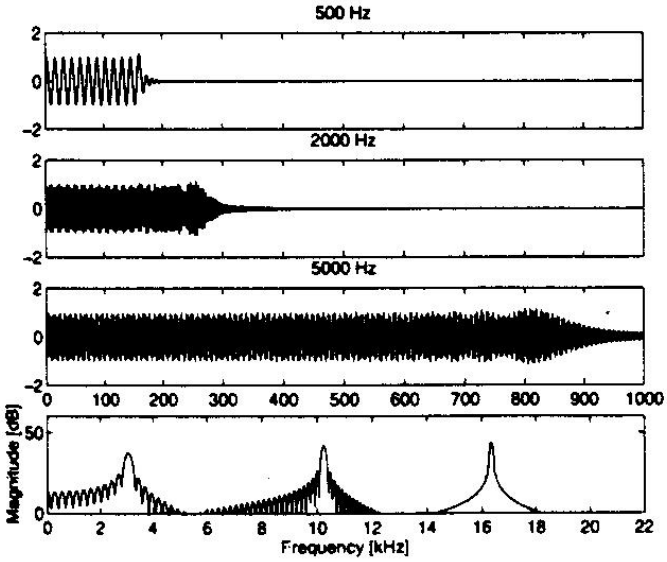

Figura 7.5: *Tres tonos en tiempo y frecuencia tras realizar un procesado* warped *por una cadena de 1000 bloques paso todo [Harma00a].*

en la escala que escucha el oído. Una cuestión no abordada hasta ahora es la determinación del número de polos necesarios en un sistema *warped*-lpc. Para este fin, se puede utilizar la ganancia de predicción [Markel76] como valor para decidir en el codificador qué orden es necesario para modelar el residuo. La ganancia de predicción asociada a un polo indica la importancia de este polo en la minimización del error. Cuanto más grande es la ganancia, mayor será la disminución del error de predicción. Además, es posible calcular la ganancia de predicción directamente a partir de los coeficientes de la estructura en celdas. Usando la ganancia de predicción para un modelo *warped*-lpc, se determina el número de polos usando un criterio perceptual, ya que se extraen los polos que minimizan el error en banda de Bark (o logarítmica) hasta un determinado umbral.

7.2.3. Comparación de resultados

A continuación, se van a realizar unas pruebas subjetivas para realizar una comparación entre métodos de modelado de ruido que siguen principios perceptuales. En principio, tanto un banco de filtros ERB como un modelo *warped*-LPC utilizan el mismo principio perceptual, el oído escucha en una escala logarítmica en frecuencia. También ambos métodos has sido utilizados en diferentes codificadores paramétricos de audio. En cuanto a implementación, quizás el banco de filtros ERB mediante FFT es el método menos complejo, aunque el *warped*-LPC también tiene una complejidad reducida [Harma00a]. Desde el punto de vista de codificación, ambos métodos pueden utilizar herramientas con base psicoacústica para limitar el régimen binario. Para el caso del banco de filtros ERB, la energía de una determinada banda se puede no enviar o cuantificar con un número de bits variable en función de la máscara de ruido actual. Para el caso *warped*-LPC, la ganancia de predicción es el valor umbral a utilizar para decidir cuántos polos debe enviar el codificador. Así pues, sólo falta conocer cuál de los métodos consigue un modelo de ruido que ofrezca mayor calidad con las herramientas de codificación de audio propuestas en esta

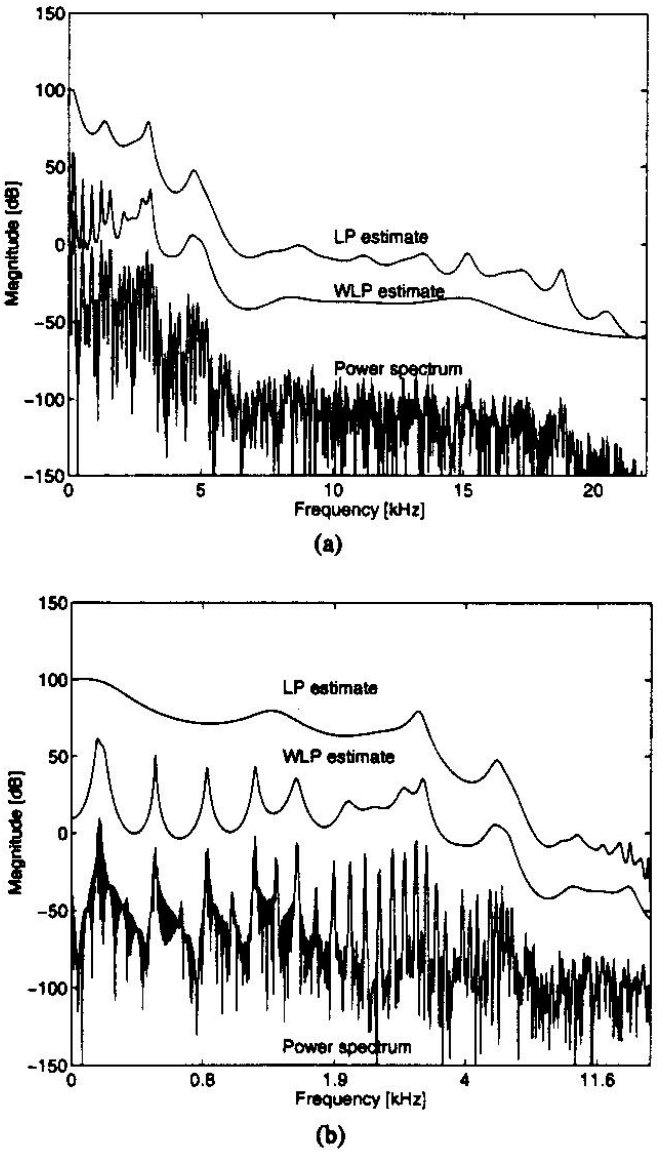

Figura 7.6: *Espectro de una señal musical de clarinete y espectro estimado por modelos LPC y* warped-*LPC de orden 40. (a) Frecuencia lineal, (b) frecuencia logarítmica [Harma00a].*

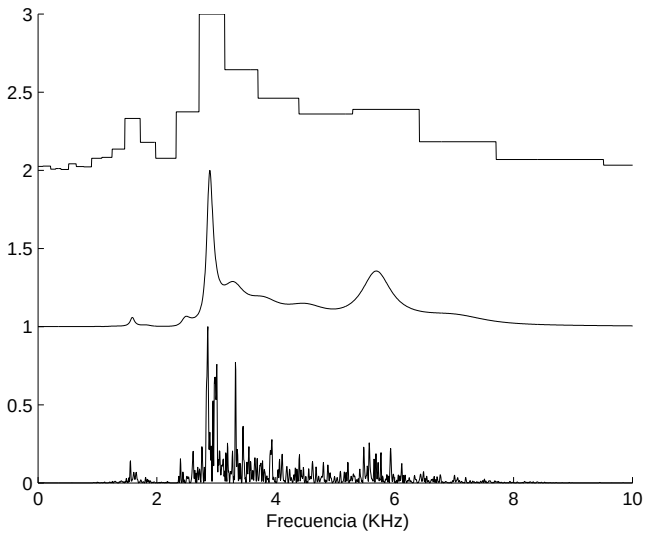

Figura 7.7: *Espectro del residuo de una señal vocal sorda (abajo), la envolvente de energía mediante* warped-*LPC con 30 polos (medio) y un banco de filtros ERB mediante FFT con 30 bandas (arriba).*

tesis. Para ello, se realizará una prueba subjetiva y se analizarán los resultados.

En primer término, se va a comparar gráficamente la envolvente de energía en frecuencia que obtienen ambos modelos de ruido para el caso de algunos segmentos particulares. En la figura 7.7 se dibuja el residuo que queda tras aplicar el modelo tonal extrayendo sólo los tonos perceptualmente importantes a un segmento sordo de señal de voz. En este caso, el residuo no tiene un pitch definido, aunque sí una forma en frecuencia que dista mucho de ser la de ruido blanco. El modelo con *warped*-LPC obtiene una forma más exacta en bajas frecuencias, ya que se puede apreciar cómo modela la poca energía que hay en baja frecuencia y, sin embargo, modela de forma más burda la mayor energía de alta frecuencia. En su caso, el modelo basado en ERB tiene anchos de banda mayores para las altas frecuencias. Con este residuo, no se puede apreciar, a priori, qué modelo obtendrá mejores resultados perceptuales.

El siguiente paso es ver los resultados del residuo para una señal musical. En el caso de la figura 7.8, el residuo en un segmento de una señal de orquesta. Ahora, el residuo tiene un espectro más coloreado aún, puesto que en esta señal no hay una fuente de naturaleza ruidosa como en la voz sorda. Así pues, el residuo está formado por los tonos no extraídos por el modelo tonal. Se observa como el modelo *warped*-LPC obtiene una envolvente de energía que modela muy bien los tonos de baja frecuencia presentes en el residuo, siendo peor el modelo para las altas frecuencias. Sin embargo, en el caso del modelo basado en ERB, no se podrán sintetizar tonos bien definidos, lo que con seguridad repercutirá en la calidad de la señal de ruido sintético.

Por último, se trata el caso del residuo para un segmento de señal sonora de voz. La figura 7.9 representa este caso. Ahora, es claramente apreciable que el pitch de la señal sonora se manifiesta en el residuo. El espectro está formado por un conjunto de tonos armónicos, aunque el modelo tonal haya extraído los tonos de baja frecuencia. Aunque el modelo *warped*-LPC intente modelar el tono más importante en baja frecuencia, es incapaz de obtener una envolvente que recupere una señal con carácter armónico. Parece obvio que no se puede pedir este resultado a un modelo

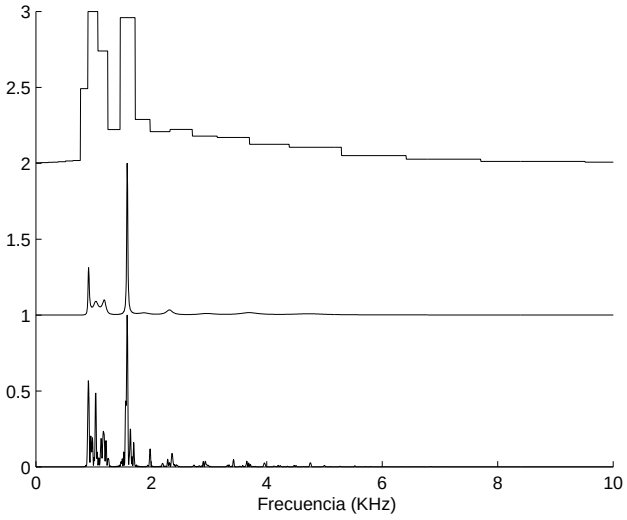

Figura 7.8: *Espectro del residuo de una señal orquestal (abajo), la envolvente de energía mediante* warped-*LPC con 30 polos (medio) y un banco de filtros ERB mediante FFT con 30 bandas (arriba).*

de ruido. La envolvente del modelo ERB tampoco obtiene un resultado mejor.

De las gráficas expuestas, se puede llegar a la conclusión que, debido a que en el residuo hay en realidad una parte de señal tonal, cualquiera que sea el modelo empleado no podrá obtener una señal similar en frecuencia al residuo. Sin embargo, si se tiene que decidir, a partir de las gráficas, cuál es el mejor modelo, todo parece indicar que el enfoque *warped*-LPC obtendrá unos mejores resultados perceptuales, aunque esta afirmación se verificará realizando un test subjetivo de comparación de ambos modelos.

El test subjetivo realizado se ha implementado extrayendo con el modelo tonal los tonos perceptualmente importantes. También se ha usado, cuando ha sido necesario, el modelo de transitorios. El residuo obtenido se ha modelado con las dos versiones propuestas de modelos de ruido con características psicoacústicas. La señal evaluada se ha obtenido sumando a los tonos y transitorios sin cuantificar el ruido sintético de cada modelo en cuestión. De esta forma, se consigue que sólo los errores del modelo de ruido aparezcan en la seña evaluada. Los resultados de preferencia de la señal obtenida con ambos modelos aparecen en la tabla 7.1.

En base a las opiniones de los 10 oyentes que han realizado el test, se puede considerar en general que el modelo *warped*-LPC se escucha menos ruidoso que el modelo ERB. En algunos casos, se escucha el efecto pre-eco más intenso en el ruido obtenido mediante *warped*-LPC, lo que hace cambiar en algunos casos la tendencia general del test. Básicamente, la explicación a los resultados obtenidos hay que buscarla en la naturaleza poco ruidosa del residuo en algunas situaciones, lo que hace difícil la obtención de una señal natural con un modelo de ruido. El modelo *warped*-LPC es más robusto a esta situación desfavorable. La minimización de estos problemas parece estar en el camino de introducir una excitación multipulso en el modelo de ruido, cuando sea necesario [Ding97].

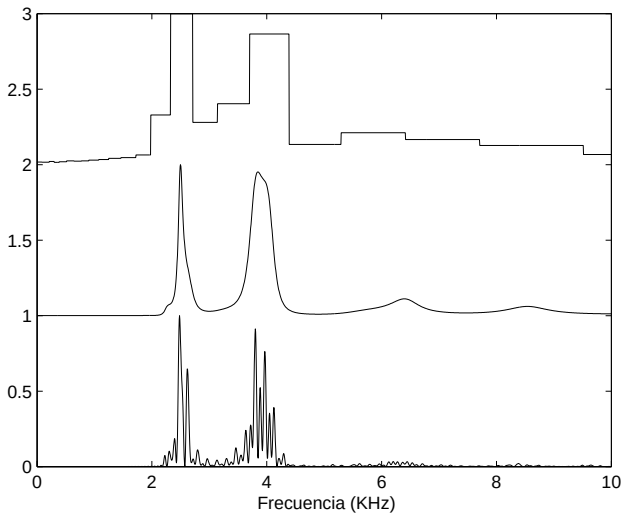

Figura 7.9: *Espectro del residuo de una señal orquestal (abajo), la envolvente de energía mediante* warped-*LPC con 30 polos (medio) y un banco de filtros ERB mediante FFT con 30 bandas (arriba).*

Cuadro 7.1: *Preferencia de los resultados del modelo de ruido WLPC sobre el modelo ERB basado en FFT en %.*

Fichero	Preferencia (%)
Suzanne Vega	80
Voz masculina en alemán	70
Voz femenina en inglés	30
Clavicordio	80
Castañuelas	50
Diapasón	100
Gaita	60
Glockenspiel	100
Punteos de guitarra	90
Sólo de trompeta	100
Pieza orquestal	70
Pop	70

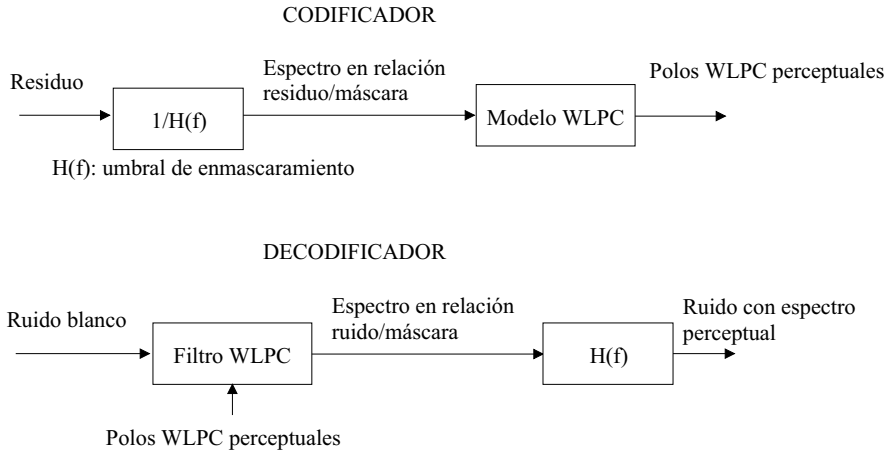

Figura 7.10: *Obtención del modelo de ruido con un espectro pesado perceptualmente gracias al umbral de enmascaramiento presente tanto en el codificador como en el decodificador.*

7.3. El espectro perceptual del ruido

Una propuesta comentada por el Dr. Antonio Pena (profesor titular de universidad de la Universidad de Vigo) realizada durante la celebración de la 118ª convención del AES en Barcelona (2005) es modelar la envolvente en frecuencia teniendo en cuenta el umbral de enmascaramiento de ruido. Esto es posible en el codificador propuesto en esta tesis, porque gracias al método propuesto de cuantificación de amplitudes de las frecuencias [Vera04b] se ha de calcular un sencillo umbral de enmascaramiento en el decodificador. Por lo tanto, es posible conocer el mismo umbral de enmascaramiento en codificación y decodificación, algo único en un codificador paramétrico de audio.

La idea es: antes de extraer los polos del WLPC, dividir en frecuencia la energía del residuo entre el umbral de enmascaramiento. De esta forma, el modelo de ruido no parametriza la envolvente del residuo, si no la envolvente de la relación residuo-máscara. Gracias a este pre-filtrado del ruido, se dedicarán más polos y, por lo tanto, se modelizará una envolvente más ajustada a aquellas zonas de frecuencia donde la relación residuo-máscara es mayor y viceversa. Como resultado esperado, se debe obtener un ruido con menor error perceptual (ya no de energía) con respecto al residuo original. Por tanto, esta es una forma de ajustar los polos del modelo de manera perceptual, y no por energía, como se hace de forma general.

Si se dibuja un diagrama de bloques de las operaciones realizadas tanto en el codificador como en el decodificador para obtener este espectro perceptual de ruido, hay que incluir un filtrado cuya respuesta en frecuencia sea el umbral de enmascaramiento (o su inverso). En la figura 7.10 se muestran las operaciones realizadas para obtener un espectro perceptual de ruido. Como se observa en la figura, el uso del umbral de enmascaramiento se limita a una ecualización previa al cálculo de los polos con el objetivo de que el modelo WLPC minimice la importancia perceptual (en lugar de la energía) en el cálculo de los polos. Este cambio se deshace en el decodificador, donde se obtiene al final un modelo WLPC pesado perceptualmente en lugar de por energía.

Para comprobar la validez del método propuesto, se analiza la señal de audio extrayendo transitorios y tonos e integrándolos en la señal de prueba intactos, mientras que el residuo

Cuadro 7.2: *Preferencia de los resultados del modelo de ruido WLPC pesado perceptualmente sobre el modelo WLPC tradicional pesado por energía en %.*

Fichero	Preferencia (%)
Suzanne Vega	0
Voz masculina en alemán	20
Voz femenina en inglés	10
Clavicordio	60
Castañuelas	30
Diapasón	70
Gaita	40
Glockenspiel	90
Punteos de guitarra	60
Sólo de trompeta	70
Pieza orquestal	0
Pop	20

resultante es modelado mediante WLPC con y sin pesado perceptual. Así pues, la señal de prueba se obtiene sumando los tonos y transitorios extraídos sin cuantificar y el ruido modelado. En esta señal de prueba es obvio que sólo se pueden apreciar los artefactos auditivos resultantes de la sustitución del residuo original por el modelo de ruido sintético. Los resultados obtenidos son basante interesantes y se muestran en la tabla 7.2. En esta tabla se presenta la preferencia de los 10 oyentes de la señal modelada con WLPC pesado perceptualmente sobre el pesado por energía.

La preferencia de la señal de ruido modelada con WLPC pesado perceptualmente no es general para todas las señales, ni siquiera mayoritaria. A partir de las opiniones dadas por los oyentes que realizaron la prueba, se puede llegar a la conclusión de que el modelo de ruido perceptual es mejor para aquellas señales que se escuchan ruidosas con el modelo WLPC directo. Sin embargo, el otro artefacto presente en el ruido, la interpretación del oído de la señal tonal como independiente de la señal ruidosa o efecto de eco, se amplifica en el modelo de ruido perceptual. Esto hace incluso desagradable el modelo para ciertas señales, como las señales habladas y la señal de pieza orquestal. Quizás los resultados se vean mejorados por la generación en el decodificador de ruido multipulso, en lugar de ruido blanco, de forma que se sinteticen algunas de las características originales del residuo, como por ejemplo la existencia de un pitch definido en las señales más críticas. De cualquier forma, ésta es una línea futura de investigación. Como artefacto adicional, las señales generadas a partir de ruido WLPC pesado perceptualmente se escuchan con menor riqueza en altas frecuencias. Este efecto de filtrado es determinante en la elección de la señal preferida, por ejemplo en el caso de la señal pop. Finalmente, en lo que a los propósitos de esta tesis se refiere, se descarta el uso del modelo WLPC perceptual porque los resultados no son del todo satisfactorios e introduce una complejidad adicional en el decodificador. Esta última razón hace recomendable la solución adoptada con el objetivo final de conseguir una decodificación en tiempo real.

7.4. La envolvente del ruido en el tiempo

En los modelos de ruido de codificadores paramétricos de audio también se modela la envolvente de la energía en el tiempo en cada trama de ruido. Normalmente, la envolvente temporal del residuo se modela mediante la energía en tiempo corto, es decir, calculando la energía cada conjunto pequeño de muestras. El único problema de diseño que hay que tener en cuenta es que la envolvente temporal debe estar adaptada a la resolución temporal del sistema auditivo humano. Así pues, es necesario determinar el tiempo máximo para actualizar la energía en tiempo corto. Este valor es muy variable según los autores. En [Brinker02], se propone un valor de $8ms$, mientras que en [Verma99] y [Purnhagen98] se admite un valor de $32ms$. Las señales que limitan este valor son las señales vocales, que tienen un fuerte carácter no estacionario y, por tanto, necesitan una rápida actualización de la energía del residuo. Algunos experimentos [Myburg04] indican que la señal de voz codificada con un codificador paramétrico se escucha metálica incluso con valores de $10ms$, aliviándose este problema con $8ms$. Con este valor, es necesario enviar la potencia de ruido cada 353 muestras para una frecuencia de muestreo de $44,1kHz$.

Si se envía la potencia de ruido cada $8ms$, el modelo obtenido está bastante limitado, porque no es posible obtener una potencia de ruido que varíe de forma natural, sino que variará en intervalos rectangulares de bastantes muestras. Por esta circunstancia, en los codificadores paramétricos de audio, se suele enviar la energía en tiempos más pequeños aún, realizando un sub-tramado adicional para conseguir una envolvente de mejor calidad [Schijndel99]. Por ejemplo, en [Myburg04], se utiliza un valor de sub-trama de ruido de $1,6ms$. Sin embargo, cuando la señal a codificar es una señal de audio, muchas veces la envolvente de ruido tiene un carácter más estacionario. Con este esquema de cálculo de la envolvente se derrocha régimen binario en algunas señales de audio para conseguir buena calidad en las señales vocales. Una estrategia de ahorro de bits se puede incluir en el cálculo de la envolvente temporal de ruido. Esta herramienta debería decidir si una trama de señal tiene o no una envolvente significativa y, en este caso, permitir el modelado en sub-tramas.

Una forma alternativa de conseguir un buen modelo de la envolvente de ruido con un bajo régimen binario de codificación es utilizar un filtro predictor basado en LPC de la transformada de la señal de ruido. Esta idea se ha utilizado con éxito en los codificadores de forma de onda para establecer una forma temporal apropiada al error de cuantificación y evitar efectos de pre-ecos en partes transitorias de señal. Este esquema conocido como TNS (*Temporal Noise Shaping*) ha sido usado con éxito en MPEG-AAC. La base teórica de este predictor se fundamenta en realizar la predicción sobre las muestras de la FFT del ruido, consiguiéndose de esta forma un filtro LPC en frecuencia cuya respuesta temporal tiene la forma de la envolvente del residuo. Con un filtro LPC en frecuencia, es posible determinar si la envolvente de la señal es significativa, a partir de la ganancia de predicción de cada polo extraído. Usando un umbral para la ganancia de predicción por encima del cual enviar los polos, se puede implementar un bloque que modele la envolvente sólo en aquellos casos en que sea necesario. En la figura 7.11 se representa la envolvente con sólo 3 polos del residuo para una señal sorda de voz. Se puede apreciar como se obtiene una envolvente con una forma suave, y no con la forma escalonada que se obtendría mediante la energía en tiempo corto.

En el codificador propuesto en esta tesis se va a utilizar un filtro LPC en frecuencia para modelar la envolvente de residuo en el tiempo. Para la cuantificación de los polos, se utilizarán

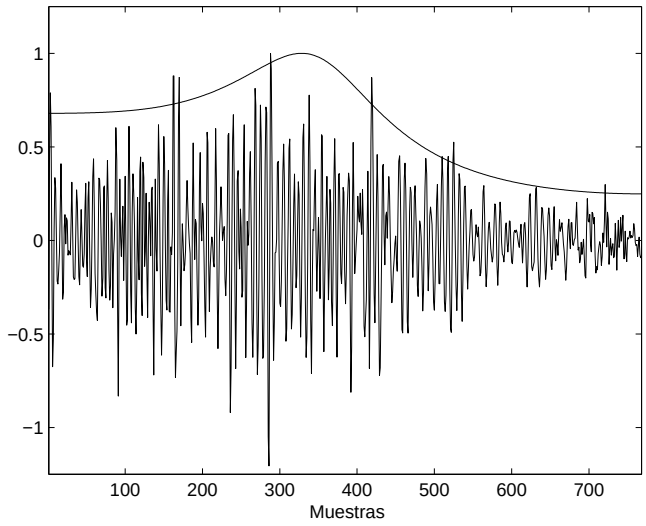

Figura 7.11: *Residuo para una señal de voz sorda y envolvente calculada con un filtro LPC en frecuencia con 3 polos.*

los coeficientes de la estructura *lattice* con un doble objetivo: por un lado, evitar inestabilidades al cuantificar y, por otro, usar un valor máximo del cuantificador de polos a uno. El número de polos a enviar se determinará mediante la ganancia de predicción de cada polo. De esta forma, se evita enviar una envolvente en aquellos casos donde no sea necesario. Usando un umbral suficientemente bajo, se puede conseguir una resolución temporal suficiente para satisfacer las exigencias del sistema auditivo.

Capítulo 8

Codificador paramétrico propuesto

El objetivo principal de esta tesis es el diseño e implementación de un codificador paramétrico de audio, a partir de las contribuciones realizadas en los diferentes modelos de señal revisados. Sin embargo, antes de empezar a justificar su estructura, es preciso dejar claros los principios que se han seguido en el diseño del codificador propuesto.

El codificador paramétrico propuesto debe operar a bajo régimen binario con la mejor calidad perceptual posible. Las posibles aplicaciones de este codificador son varias, aunque quizás la más prometedora es su uso para *streaming* de audio por internet. Se ha diseñado, por tanto, este codificador teniendo en cuenta que los datos codificados pueden ser enviados por una red de transmisión de paquetes (y no por un canal dedicado), lo que puede suponer la pérdida de algún paquete completo en aplicaciones de tiempo real. Así pues, la información que genera el codificador propuesto será completamente independiente entre segmentos, lo que limita en cierta medida la capacidad de compresión. Por esta razón, no se usará codificación diferencial inter-trama, aunque sí se puede usar codificación diferencial intra-trama. Para comprender la limitación que introduce este principio de diseño, se puede poner un ejemplo con las frecuencias de los tonos. En la mayoría de los codificadores de audio, se utiliza el seguimiento de caminos tonales como una herramienta de codificación. De esta forma, se consiguen reducciones de más del 50 % en los bits dedicados a las frecuencias mediante el empleo de codificación diferencial entre frecuencias que pertenecen al mismo camino tonal y que corresponden a tonos que se extienden durante varios segmentos [Levine98]. Otro motivo que induce al codificador propuesto a la codificación independiente entre segmentos es la escalabilidad. La línea de investigación más importante que se dibuja a partir de esta tesis doctoral es la implementación de un codificador paramétrico escalable. Si este codificador escalable realiza una codificación independiente entre segmentos, podrá modificar instantáneamente el régimen binario sin ningún cambio en la codificación de los parámetros. Como conclusión, en el codificador propuesto se evita cualquier estrategia de codificación diferencial entre tramas, aunque se han intentado estrategias de codificación intra-trama.

Otro principio de diseño derivado de la aplicación de *streaming* es que el régimen binario puede ser variable. Como el codificador propuesto no se diseña para ser utilizado en un canal dedicado con un régimen binario definido, se puede codificar la señal con un régimen binario variable. Gracias a esta propiedad, se utilizará un mayor régimen binario para aquellos segmentos de señal que lo necesiten. La complejidad de implementar un codificador de audio a régimen binario

variable es menor, ya que no se implementan estrategias de control del régimen binario. Sin embargo, esto no quiere decir que se vaya a conseguir una calidad transparente. Los errores propios de los modelos paramétricos repercutirán en la señal de salida. Como ejemplo, es típico escuchar en la señal codificada pitidos debido a tonos que se extraen en zonas de ruido o señales ruidosas por modelar como ruido partes tonales de la señal. Estos errores son inherentes al uso de modelos paramétricos e inevitables desde el punto de vista de cuantificación.

8.1. Estructura del codificador de audio propuesto

Como se verá a continuación, un codificador paramétrico debe contener, además de los modelos de señal, una serie de algoritmos de control para el procesado de la señal de audio. Estos algoritmos son básicamente un bloque que realice la segmentación de la señal de entrada y un detector de transitorios. La estructura del codificador de audio propuesto debe incluir otros algoritmos adicionales para el tratamiento del ruido. Así, es normal en un codificador paramétrico enviar sub-tramas de ruido y usar un detector de micro-transitorios. Por lo tanto, las herramientas de control a definir son:

Segmentación. Para aprovechar al máximo los modelos de señal e incrementar la capacidad de compresión del codificador, es preciso definir un bloque que implemente la segmentación de la señal de audio. El objetivo es separar en segmentos las partes estacionarias de la señal de audio. Por ejemplo, para el caso de la señal de voz, es muy interesante dejar en el mismo segmento la señal de un determinado fonema, cambiando rápidamente de segmento cuando éste se cambia. Lo mismo ocurre para otras señales de audio. En una señal musical, es recomendable tratar en el mismo segmento la misma nota y cambiar de segmento cuando ésta se cambia. En principio, las partes transitorias de señal no es necesario separarlas en muchos segmentos pequeños, porque el modelo de transitorios extrae los cambios bruscos de señal evitando el pre-eco. Básicamente, los segmentadores que aparecen en la bibliografía trabajan a partir de los cambios espectrales de la señal de entrada. En el apartado 8.2 se realiza una descripción del segmentador implementado.

Detector de transitorios. Para incluir o no el modelo de transitorios en la cadena de modelos de señal, es necesario implementar un detector que decida sobre esta cuestión. Los detectores de transitorios son herramientas sencillas basadas en comprobar fuertes incrementos de la energía de la señal de entrada. En el apartado 8.3 se realiza una breve descripción del detector utilizado.

Detector de micro-transitorios. Los micro-transitorios son pequeños golpes de algunos instrumentos presentes en la señal, pero que no contienen la mayor parte de la energía de la misma. La señal de la figura 6.13 es un claro ejemplo de micro-transitorio. Estos micro-transitorios no son detectados por las herramientas típicas de detección de transitorios basadas en energía. La única forma de que no se modelen como ruido es analizar el residuo del modelo tonal para comprobar si tiene algún cambio brusco de energía [Levine98]. Si hay un micro-transitorio, se aplica un modelo de transitorios para modelar esta señal.

Segmentación de sub-tramas para el ruido. El segmento o trama de audio usada en los codificadores paramétricos es demasiado grande generalmente para usarla en el modelo de

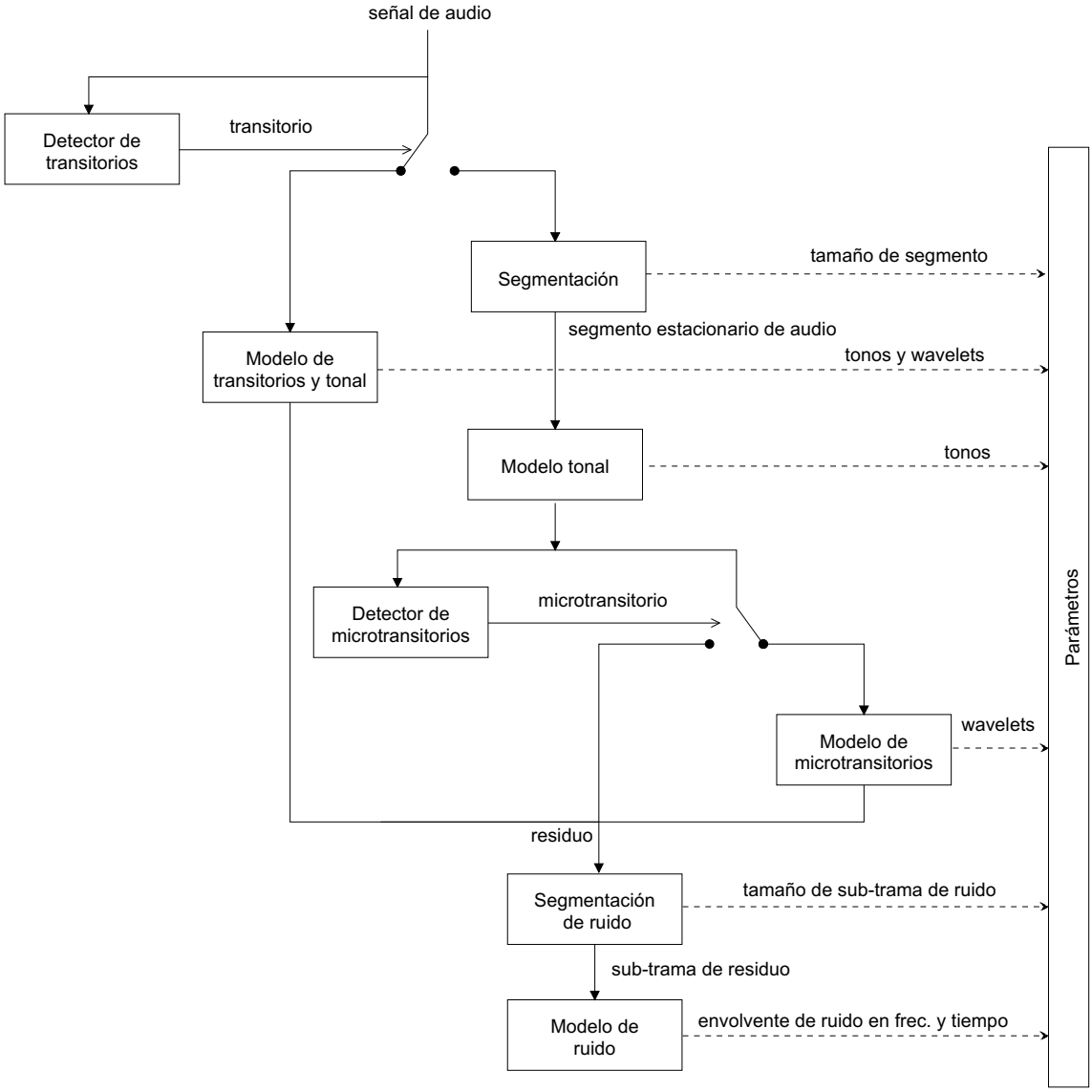

Figura 8.1: *Estructura del codificador paramétrico propuesto.*

ruido. Es común, por lo tanto, realizar una segmentación adicional del ruido [Verma99] [Myburg04], aunque parece lógico no poner un tamaño de trama fijo y lo suficientemente pequeño como para modelar bien todos los posibles residuo. La solución propuesta se basa se realizar un segmentador de ruido que divida el residuo en tramas de tamaño variable donde se agrupe el residuo con propiedades estacionarias.

Con estas herramientas de control (cuya implementación se comentará posteriormente) y con los modelos de señal descritos hasta ahora, se puede implementar un codificador completamente paramétrico de audio con unos resultados prometedores. La estructura del codificador paramétrico de audio propuesto se representa en la figura 8.1. Como se observa, además de las herramientas de control nombradas arriba, se han introducido los modelos de señal para tonos, transitorios y ruido. Analizando la figura, se pueden dar tres circunstancias de funcionamiento:

1. Segmento transitorio. En este caso, se activa el detector de transitorios pasando la señal por el modelo conjunto que extrae tonos y transitorios. El residuo producido se pasa directamente al segmentador de ruido.

2. Segmento estacionario. En este caso, se aplican en serie el segmentador para decidir el tamaño del segmento estacionario y el modelo tonal en dicho tamaño de segmento. En ausencia de micro-transitorios, el residuo del modelo tonal se modelará como ruido.

3. Segmento con micro-transitorio. Si en un segmento estacionario, tras el modelo tonal, se detecta un micro-transitorio, se aplica un modelo de transitorios a la señal residual del modelo tonal. El residuo final se pasa al modelo de ruido.

Tras analizar las tres clases de segmentos, se comprueba que, aunque en la figura 8.1 haya tres ramas que confluyan en el segmentador de ruido, sólo una de ellas está activa a la vez. Por lo tanto, sólo hay un residuo para ser modelado por segmento.

En relación a las herramientas utilizadas en cada uno de los bloques, se tiene la siguiente descripción:

Modelo de transitorios y tonal. Se ha implementado el algoritmo *matching pursuits* con un diccionario mixto de funciones wavelet-packets y exponenciales complejas. En este diccionario se han utilizado $L = 4097$ frecuencias y una descomposición wavelet-packets de profundidad $P = 4$ con filtros de 32 coeficientes de la familia ortonormal de *Daubechies*. El algoritmo *matching pursuits* se detiene cuando un átomo extrae menos del $1,75\,\%$ de la energía del residuo en esa iteración. Este valor, elegido de forma heurística, se escoge para que el residuo tenga características ruidosas y no conserve transitorios de señal que provoquen errores de modelado. El tamaño de la longitud de la trama se tratará posteriormente en el apartado 8.2.

Modelo tonal. Se usa aquí *Perceptual Matching Pursuits* con criterio de parada perceptual. El número de frecuencias es de $L = 4097$ y se detiene el algoritmo cuando se han extraído todos los tonos perceptualmente importantes. El número de frecuencias se elige para obtener una buena discriminación en frecuencia, mayor que el JND para casi toda la frecuencia. La máscara necesaria para evaluar la importancia perceptual de cada tono se calcula con el modelo de enmascaramiento 2 de MPEG [MPEG92], modificado para calcular la máscara sobre los tonos, sin incluir el efecto tono sobre tono. Así, se obtiene la máscara de ruido sobre tono más umbral de silencio necesaria en PMP. En cuanto al enventanado, se trabajará con ventana rectangular en análisis y trapezoidal en síntesis (para suavizar la transición entre tramas). Por lo tanto, en el codificador propuesto no habrá solapamiento entre segmentos en análisis, aunque, debido a la ventana trapezoidal en síntesis, sí que lo habrá en el decodificador. En concreto, se utiliza un solapamiento del $10\,\%$ entre ventanas trapezoidales. Este enfoque, que minimiza el régimen binario, se utilizará en el resto de modelos de señal.

Modelo de micro-transitorios. El algoritmo *matching pursuits* se implementa ahora con un diccionario de funciones wavelet-packets. No se incluyen las funciones tonales, porque cuando el residuo llega a este modelo éstas ya se han extraído. Las funciones wavelet-packets

se basan en filtros de 32 coeficientes de la familia ortonormal de *Daubechies* y una descomposición WP de profundidad $P = 4$. El algoritmo se detiene con el mismo umbral de energía que en el caso del diccionario mixto.

Modelo de ruido. Una vez extraídas las sub-tramas de ruido, en cada una de ellas se obtienen los parámetros que modelan la envolvente de energía en frecuencia y tiempo. La envolvente de frecuencia se obtiene con un filtro predictor basado en *Warped*-LPC. El número máximo de polos se limita en función del tamaño de la sub-trama de ruido. Esto se consigue usando como máximo 1 polo por cada 32 muestras de residuo. El número de polos no se fija a este valor, sino que si los polos tienen una ganancia de predicción menor de $0,01$ no son codificados. La verificación de los polos que tienen una ganancia de predicción mayor que el umbral se realiza empezando por el final. Una vez calculados todos los polos, partiendo del último, se verifica que estén por encima del umbral, en caso contrario, son eliminados. Así pues, se van eliminando polos hasta encontrar el primero por la cola con una ganancia mayor que el umbral. Esto se hace así porque los polos no se obtienen en orden decreciente de ganancia de predicción y se pueden eliminar polos importantes. En cuanto al modelado de la envolvente temporal, se utiliza un predictor LPC de las muestras de la FFT del ruido. También se usa un polo cada 32 muestras y una ganancia de predicción mínima de $0,01$.

Todos los parámetros generados por el modelo deben ser cuantificados, si es posible siguiendo principios psicoacústicos. En este sentido, se generan tres tipos de parámetros:

- Parámetros que se pueden cuantificar usando directamente criterios perceptuales. Las amplitudes de los tonos, por ejemplo.
- Parámetros que se cuantifican directamente sin información perceptual. Las fases de los tonos son un buen ejemplo.
- Parámetros que son discretos por la naturaleza del modelo empleado. Las frecuencias de los tonos son discretas porque hay tantas frecuencias como átomos exponenciales complejos use el algoritmo *matching pursuits*.

Por lo tanto, en función del parámetro en cuestión, se usará un tipo diferente de cuantificador. Antes de pasar a este punto, se verán con más detalle los algoritmos empleados para segmentación del eje temporal y detección de transitorios.

8.2. Segmentación del eje temporal

Desde un punto de vita teórico, el tamaño de segmento óptimo para un codificador de audio debe ser aquel que consiga minimizar el régimen binario obteniendo una buena calidad en la señal de audio codificada [Prandoni97]. Sin embargo, en un segmentador de este tipo sólo se puede conocer a posterior el régimen binario obtenido, lo que es prohibitivo en términos de complejidad computacional.

Como este objetivo tan ambicioso no es posible, en codificación de audio se utilizan herramientas de procesado de señal que segmentan el audio en función de las características propias de cada codificador [Painter01]. Así, en codificadores de forma de onda, es típico usar un tamaño

de trama grande en segmentos estacionarios de señal, reduciéndolo en zonas transitorias. La explicación de este hecho se debe a que se produce una importante distorsión de pre-eco si se cuantifica un segmento largo de señal transitoria. Las herramientas usadas para decidir si un segmento es estacionario o transitorio (y decidir en base a esto la segmentación) deben ser sencillas. En [Gonzalez01], se usa la transformada wavelet-packet como herramienta de tratamiento. Independientemente de la herramienta utilizada, un segmentador debe partir la señal analizando tanto los cambios de energía en frecuencia como los cambios de energía en el tiempo presentes en la señal. Se ha demostrado que midiendo distancias sencillas en base a los cambios de energía en tiempo-frecuencia se puede conseguir un algoritmo eficaz para segmentación [Ruiz02].

En lo que a codificación paramétrica de audio se refiere y con las herramientas utilizadas, los principios del segmentador cambian por completo. Ahora, los segmentos transitorios de señal no se deben partir, puesto que la herramienta de modelado de transitorios no provoca pre-ecos en la señal. Como mucho, el residuo en un segmento transitorio deberá partirse en sub-tramas para evitar tal circunstancia. Sin embargo, el modelo tonal a utilizar asume que la señal es estacionaria para usar información perceptual en la extracción tonal. Por lo tanto, sólo se debe pasar al modelo tonal segmentos de señal estacionarios. Debe observarse también que los átomos del modelo tonal son exclusivamente funciones exponenciales complejas que no cambian en amplitud ni frecuencia en todo el segmento de señal analizado. Así pues, los requerimientos del codificador paramétrico propuesto, en lo relativo a segmentación, son muy particulares y se pueden resumir en dos puntos principales:

1. Los segmentos estacionarios deben ser segmentados con mucha precaución. El segmentador debe permitir cortar un determinado segmento en el momento en que cambien las propiedades espectrales de la señal de entrada.

2. Los segmentos transitorios no tienen limitación alguna, puesto que el modelo de transitorios no provoca efectos de pre-eco.

En base a estos requerimientos, el algoritmo de segmentación debe conocer de antemano si una señal es o no transitoria. Si es transitoria, el algoritmo de segmentación no debe partir la señal. Sin embargo, las señales estacionarias, que se van a parametrizar con el modelo tonal, deben segmentarse con cuidado. Por esta causa en la figura 8.1 el detector de transitorios decide primero si una señal es o no transitoria, y en caso negativo, el segmentador debe dividir la señal en trozos prácticamente estacionarios. Con esta premisa, el funcionamiento del segmentador se basará en la detección de cambios del contenido espectral. Cuando éstos se produzcan, resultará un nuevo segmento de señal. Los cambios de energía temporal no se tienen en cuenta en el segmentador, porque son tratados por las señales transitorias. Al fin y al cabo, al segmentador le llegan bloques no transitorios de señal.

A continuación, se cuenta el diseño del algoritmo implementado para segmentación. El algoritmo está basado en un filtro predictor *warped*-LPC, que proporcionará la información del contenido espectral de la señal. Se ha usado un filtro predictor porque, si tiene un orden bajo, la complejidad es reducida. Además, se ha utilizado la versión modificada *warped*-LPC, porque el sistema requiere mucha definición para señales estacionarias en baja frecuencia. Con esa versión, extrayendo sólo un polo de la señal, es posible implementar un segmentador con un comportamiento aceptable para esta aplicación. El uso de un sólo polo *warped*-LPC permite obtener, a

grandes rasgos, la frecuencia donde se concentra la mayor parte de la energía de la señal, considerando el eje de frecuencias con carácter logarítmico. El algoritmo diseñado se basa en medir las diferencias de esta frecuencia central entre trozos de señal para, a partir de esta información, decidir el tamaño de segmento actual. Un diagrama del segmentador usado se representa en la figura 8.2, donde se pueden apreciar las siguientes singularidades:

- El tamaño de segmento máximo es de 3072 muestras, que con una frecuencia de muestreo de $44,1kHz$ corresponde a $69,7ms$ de señal. Es un tamaño suficiente para tener una buena compresión en señales muy tonales. Normalmente, es extraño que la señal de audio sea suficientemente estacionaria con un tamaño mayor. El número de frecuencias del modelo tonal se ha elegido mayor que este valor, teniéndose por lo tanto un diccionario sobrecompleto.

- Se calcula un polo cada 512 muestras ($11,6ms$ en tiempo), siendo el algoritmo capaz de detectar cambios espectrales a partir de los polos calculados en función de este valor.

- El cálculo de las distancias en frecuencia no se aplica a la diferencia entre frecuencias consecutivas. Si la distancia se calcula de esta forma, y la frecuencia central cambia lentamente, se puede tener un segmento de gran tamaño cuyas partes inicial y final sean demasiado diferentes. Por esta causa, la distancia se mide con respecto al máximo y al mínimo de las frecuencias anteriores, es decir, si la frecuencia central actual es mayor que cualquiera de las anteriores se calcula la diferencia con respecto al mínimo de todas (o viceversa).

- Los umbrales máximo y mínimo se sitúan en el 10 % de la máxima frecuencia digital (que es la mitad de la frecuencia de muestreo) y se calculan sobre las diferencias. Estos umbrales se han obtenido para tener una discriminación suficiente en frecuencia.

- La complejidad del algoritmo se puede reducir si el polo se calcula con menos de las 512 muestras, lo cual no modifica los resultados finales. No es necesario además volver a calcular los polos en trozos de 512 muestras en donde ya se han calculado en ejecuciones anteriores del algoritmo.

Para terminar con el segmentador, se presentan las figuras 8.3 y 8.4, donde se dibujan dos casos diferentes de segmentación. En la figura 8.3 se dibuja una señal de trompeta en un cambio de nota. La línea marca el límite del segmento decidido por el algoritmo de segmentación. Se observa cómo el límite del segmento se sitúa en la zona de separación entre las notas. Por su parte, la figura 8.4 dibuja una señal de voz cuando se termina de pronunciar un fonema sonoro. Se observa cómo el algoritmo parte el segmento de forma conservadora. Es interesante tener en cuenta que esta buena selectividad en frecuencia es posible gracias al empleo de un polo *warped*-LPC, mientras que con predicción LPC no se obtiene una discriminación tan buena en bajas frecuencias.

En la estructura del codificador de la figura 8.1 hay otro algoritmo de segmentación, en este caso para calcular las sub-tramas del residuo. Si bien este algoritmo se podría calcular en función de medidas estadísticas, se ha implementado siguiendo un esquema muy similar al del algoritmo de segmentación de audio de entrada. Al igual que para el caso del segmentador de

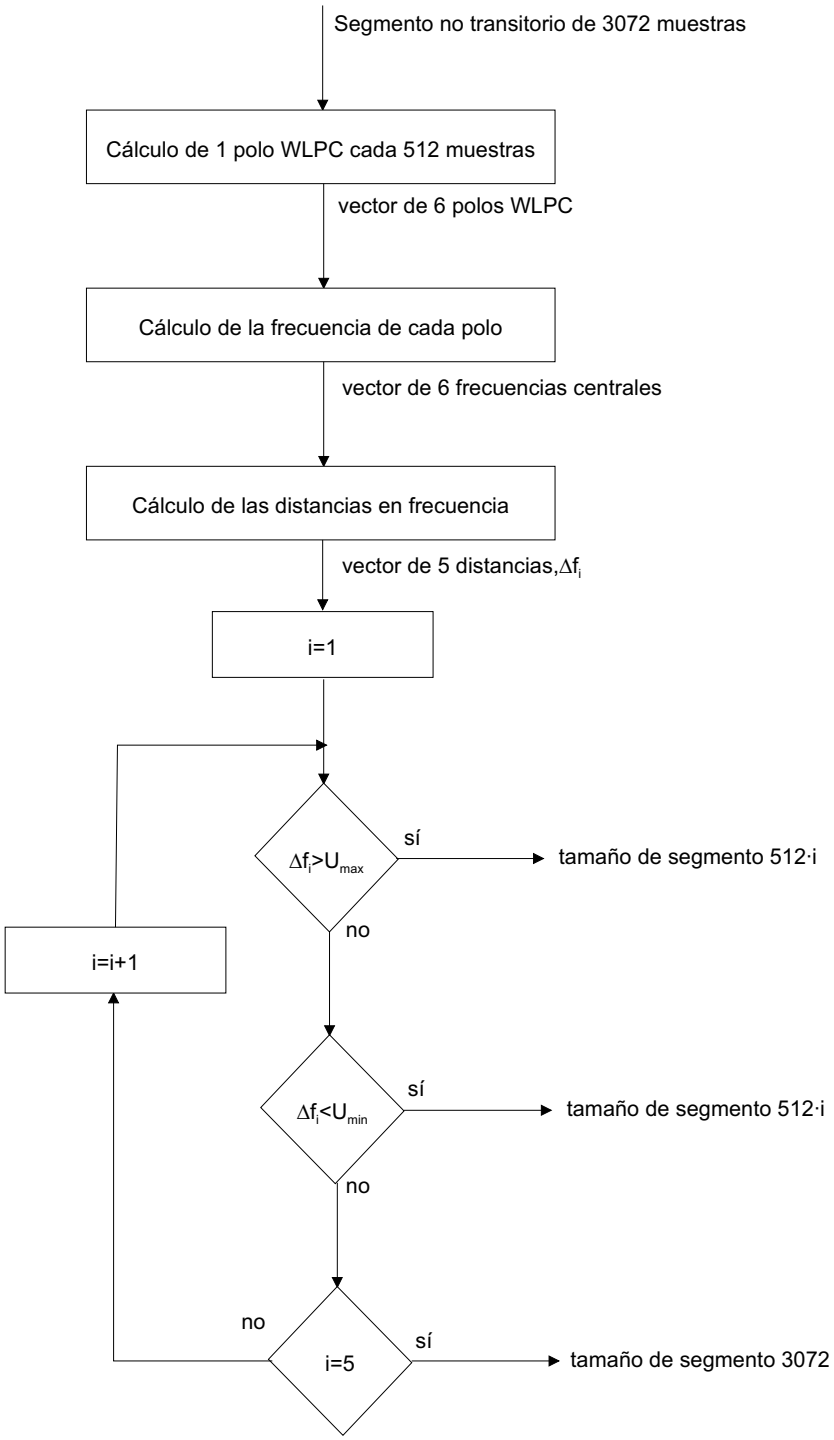

Figura 8.2: *Diagrama del segmentador usado basado en* warped-*LPC.*

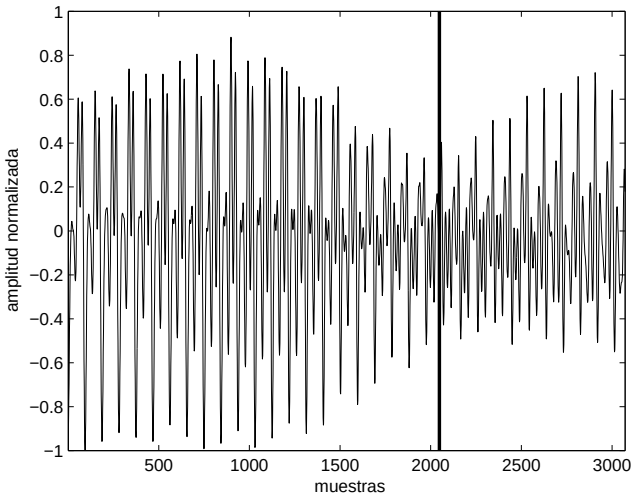

Figura 8.3: *Señal de trompeta en un cambio de nota. La línea marca el límite del segmento que calcula el algoritmo de segmentación.*

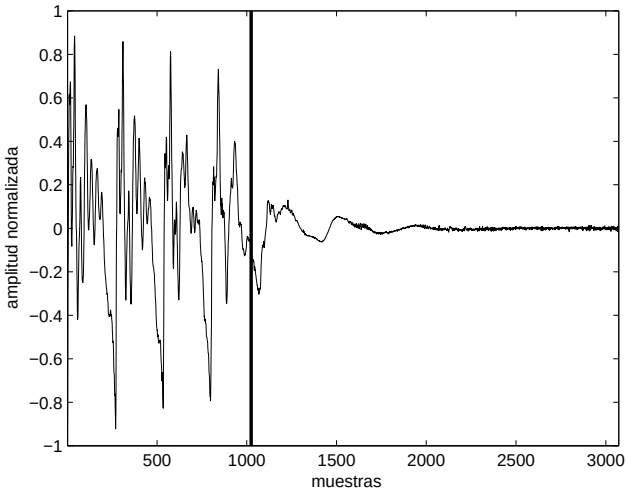

Figura 8.4: *Señal de voz cuando se termina de pronunciar un fonema sonoro. La línea marca el límite del segmento que calcula el algoritmo de segmentación.*

audio, el segmentador de ruido debe actuar cuando haya cambios locales en la energía tiempo-frecuencia. Si esto no se hace así, los parámetros que representan la envolvente de ruido en tiempo y frecuencia no modelarán de forma satisfactoria los cambios que se producen en el residuo. Hay que tener en cuenta que, en el codificador propuesto, el residuo no va a tener incrementos bruscos de energía en el tiempo, porque son extraídos por el modelo de transitorios y tonal, si es un segmento transitorio, o por el modelo de transitorios, si es un segmento estacionario con un micro-transitorio. Gracias a esta propiedad, se puede implementar un segmentador de ruido similar al de señal, aunque con las siguientes particularidades:

- Se utiliza un polo LPC, porque ya no es necesario una mejor selectividad en baja frecuencia, sino la misma para toda la frecuencia.
- El tamaño mínimo de segmento de ruido es de 256 muestras ($5,8ms$ de residuo), por lo que se calcula un polo cada 256 muestras.
- El umbral se sitúa ahora en el 15 % de la máxima frecuencia digital. Este valor se ha incrementado porque el residuo tiene más variación en frecuencia que el audio de entrada.

A continuación, se describe brevemente el funcionamiento del detector de transitorios. Este bloque es el último que queda por describir de las herramientas de análisis de señal del codificador propuesto.

8.3. Detector de transitorios

El detector de transitorios se implementa de una manera muy sencilla. Simplemente se calcula la energía de la señal a la entrada cada 256 muestras ($5,8ms$). Si se escoge un valor menor, se puede llegar a confundir un transitorio con un tono de muy baja frecuencia. Los valores mayores pueden obtener un valor demasiado promediado, evitando así la detección de un incremento local de energía. Si la energía en un trozo de 256 muestras es mucho mayor que en las zonas cercanas, se etiqueta el segmento como transitorio. De manera descriptiva, los pasos a seguir para detectar transitorios son:

1. Se analiza una señal de entrada de 3072 muestras.
2. Se calcula la energía total cada 256 muestras.
3. Si el máximo de energía de un trozo de 256 muestras es $6,5$ veces mayor que la media del conjunto de 3072 muestras, sin la contribución de dicho trozo, entonces se tiene un segmento transitorio. Este valor se escoge para modelar todos los golpes de castañuela en la señal *si*02 del grupo de señales de test EBU-SQAM.
4. En caso contrario, se pasan las 3072 muestras de señal de entrada al segmentador descrito anteriormente.

En la figura 8.5 aparece un golpe de castañuela que se detecta como transitorio.

El detector de micro-transitorios se implementa de la misma forma, salvo que el umbral es algo menor, siendo ahora de $5,5$. Este valor se escoge para modelar como micro-transitorios todos

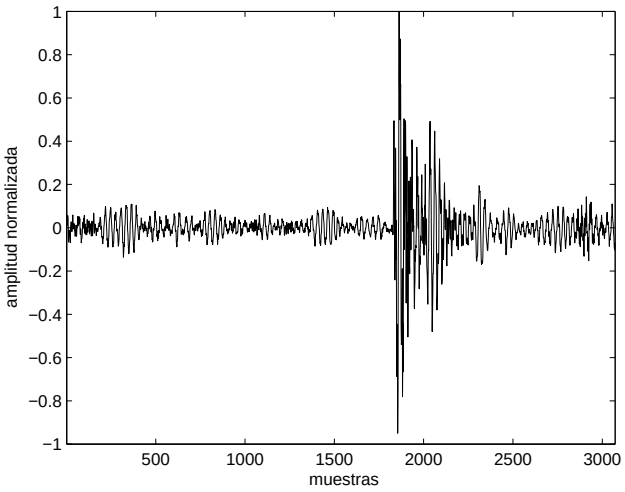

Figura 8.5: *Golpe de castañuela detectado como transitorio.*

los golpes de señal presentes en el fichero *sm*02. En la figura 8.6 se representa la señal original y el residuo del modelo tonal cuando se detecta un micro-transitorio. Es preciso aclarar que el detector de micro-transitorios analiza el residuo del modelo tonal.

8.4. Cuantificación de parámetros

Una vez que se obtienen todos los parámetros procedentes de los diferentes modelos de señal y de las herramientas de control, el siguiente paso es tratar la cuantificación y codificación de estos parámetros. El diseño de un cuantificador se define a partir de dos valores: el número de bits y el factor de sobrecarga (valores máximo y mínimo). El número de bits se debe elegir, si es posible, en función de criterios perceptuales. Para un codificador paramétrico de audio, es necesario diseñar tantos cuantificadores como tipos de parámetros se generen, por eso se realiza a continuación una revisión de los cuantificadores diseñados en cada caso.

8.4.1. Parámetros de control

En relación a las herramientas de control, el único parámetro que se codifica es el tamaño de segmento. El segmentador divide el segmento actual en un tamaño que será múltiplo de 512 muestras con un máximo de 3072 muestras. Con estos valores, el número de bits necesarios será de 3, ya que el valor mínimo es 512 muestras y el máximo 3072, siendo posibles 6 valores diferentes.

Con este valor, se inicia la codificación del segmento actual. Así pues, cada segmento es independiente del resto desde el punto de vista de codificación, habiéndose evitado por completo la codificación diferencial entre segmentos. Además, se ha intentado minimizar la dependencia del número de bits de ciertos parámetros de la información recibida de parámetros anteriores. De esta forma, aún perdiendo ratio de compresión, se hace que cada segmento enviado sea más robusto a la pérdida de sincronismo debida a errores de transmisión.

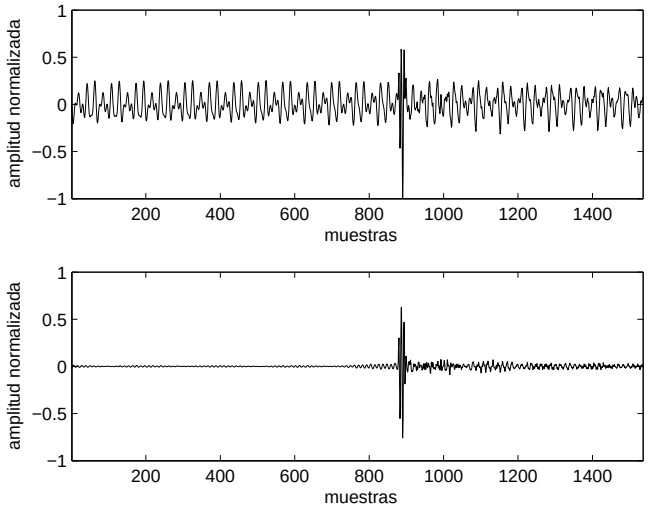

Figura 8.6: *Micro-transitorio detectado en la señal sm*02*. Se dibuja la señal de entrada (arriba) y el residuo del modelo tonal (abajo).*

8.4.2. Parámetros de los tonos

Para los tonos modelados en el codificador propuesto, los parámetros generados son amplitud, fase y frecuencia para cada tono. Con el fin de que el decodificador conozca cuántos tonos se han codificado, es necesario enviar también el número total de tonos en el segmento actual. Para cada uno de estos parámetros, se ha elegido la siguiente estrategia de cuantificación:

Número de tonos. El número de tonos se ha de cuantificar con un número de bits suficiente para evitar en lo posible que en un segmento no se pueda enviar algún tono audible por sobrepasarse el valor de diseño escogido. Bajo esta premisa, con 8 bits (hasta 255 tonos por segmento) se evita la pérdida de tonos para todos los segmentos de todas las señales de tests utilizadas.

Frecuencia. La frecuencia es un parámetro de naturaleza discreta en el codificador paramétrico propuesto. Como el número de funciones exponenciales complejas del modelo tonal es de $L = 4097$, éste es el número posible de frecuencias. Pese a que se puede enviar este valor con 12 bits (obviando la frecuencia cero), se ha implementado el codificador de frecuencias de [Ali95], explicado en el apartado 5.3.1, para ahorrar régimen binario. Este método de cuantificación de frecuencias se basa en la sensibilidad logarítmica en frecuencia del oído humano, dividiéndose el eje de frecuencias de forma diádica en 4 grupos. Se necesitan 2 bits para distinguir el grupo actual, y en la implementación realizada se ha incrementado un poco la sensibilidad del cuantificador respecto a la propuesta de [Ali95]. Se han elegido 512 escalones de cuantificación (9 bits) para cada grupo. Por tanto, se necesitan 11 bits para codificar la frecuencia de cada tono. Una ventaja de elegir el mismo número de escalones por grupo es que se evita la dependencia del número de bits de cada grupo de los 2 bits iniciales que indican el grupo actual. Esto permite que si se produce un error de transmisión en los 2 bits iniciales no se pierda el sincronismo del segmento actual en el decodificador.

Fase. Para la fase, se utiliza un cuantificador uniforme de 6 bits, al igual que en [Ali95]. Este valor está comprendido entre $-\pi$ y π. Sería conveniente como línea de futuro establecer qué tonos son sensibles a la cuantificación de la fase en base a información perceptual.

Amplitud. Las amplitudes se codifican teniendo en cuenta principios perceptuales. Se ha utilizado para cuantificar las amplitudes el mecanismo explicado en el apartado 5.3.3 [Vera04b]. Usando este algoritmo, cada amplitud tiene un número de bits variable en función de la máscara calculada en el cuantificador. Los valores máximo y mínimo son también diseñados bajo criterios psicoacústicos. Así, el valor máximo corresponde al del tono de mayor amplitud y el mínimo al de la máscara de tonos actual. Este mecanismo de cuantificación es un ejemplo de codificación intra-trama, pues explota las relaciones entre los tonos de un mismo segmento para ahorrar régimen binario. El problema que introduce es que los bits para cada amplitud son variables y dependientes de los bits de amplitud recibidos anteriormente. Esto hace muy sensible a errores esta información, porque un error de bit puede provocar la pérdida de sincronismo del segmento actual. Por esta razón, se codifica un valor adicional referente al número de bits total para la amplitud enviados en el segmento actual. Este valor informará del tamaño de la información de amplitud previniendo un error de sincronismo global en el segmento. Como el número máximo de tonos es de 256 y no es normal obtener más de 8 bits por amplitud de media, se dedican 11 bits (de 1 a 2048 valores) para informar del número de bits para las amplitudes.

8.4.3. Parámetros de las funciones wavelet-packets

En el caso de los parámetros que modelan la parte transitoria de señal, éstos no pueden ser cuantificados con criterios perceptuales. Al fin y al cabo, éstos parámetros se presentan cuando la señal no tiene un fuerte carácter estacionario, que es donde se define el proceso de enmascaramiento simultáneo. Los distintos parámetros codificados para las funciones wavelet-packets son:

Número de funciones. El número máximo de funciones por segmento se ha establecido en 256 funciones, por lo que se codifica este valor con 8 bits (entre 0 y 255). Con este valor, se evita la pérdida de funciones wavelet-packets para todos los segmentos de todas las señales de tests utilizadas.

Profundidad. Como la profundidad usada en el codificador propuesto para el árbol de descomposición wavelet-packets es $P = 4$, se necesitan 2 bits para su codificación.

Sub-banda y retardo. Se han codificado de manera conjunta estos dos parámetros con el objetivo de evitar un pérdida de sincronismo si hay un error en la profundidad de la función wavelet-packets recibida. Para ello, hay que tener en cuenta que, dependiendo de la profundidad de la función wavelet-packets actual, el número de bits para codificar la sub-banda, por un lado, y el retardo, por otro, cambian, pero en su conjunto se compensan. Tanto la sub-banda como el retardo son valores discretos en el codificador, puesto que ambos, junto con la profundidad, definen el átomo actual dentro del diccionario wavelet-packets. La sub-banda depende de la profundidad, porque cada vez que baja el nivel de profundidad el número de sub-bandas se multiplica por dos. El retardo también depende

de la profundidad, por el diezmado que se produce en el árbol de descomposición. Cada vez que se baja en profundidad, el número de retardos se reduce a la mitad. El número de bits total de la codificación conjunta sub-banda y retardo sólo depende del tamaño del segmento actual, aunque su reparto depende de la profundidad recibida. Como el tamaño de segmento varía entre 512 y 3072 muestras, el reparto de bits en función de la profundidad queda:

1. Profundidad 1. 1 bit para la sub-banda y de 8 (512 muestras para el segmento con 256 retardos a esta profundidad) a 11 bits (3072 muestras del segmento actual) para el retardo.
2. Profundidad 2. 2 bits para la sub-banda y de 7 (512 muestras para el segmento con 128 retardos a esta profundidad) a 10 bits (3072 muestras) para el retardo.
3. Profundidad 3. 3 bits para la sub-banda y de 6 a 9 bits para el retardo.
4. Profundidad 4. 4 bits para la sub-banda y de 5 a 8 bits para el retardo.

Por tanto, el número de bits total variará entre 9 a 12 bits, en función del tamaño del segmento actual e independientemente de la profundidad. Es preciso notar la sensibilidad de este parámetro codificado en función del tamaño de segmento. Un error en este valor puede provocar una pérdida de sincronismo. Como conclusión, el tamaño de segmento es un información crítica en el conjunto de la trama enviada.

Amplitud. La amplitud de cada función wavelet-packets no es un parámetro de naturaleza discreta como los anteriores, sino que hay que definir un cuantificador para su codificación. Este cuantificador no puede basarse en principios psicoacústicos, como en el caso de las amplitudes de los tonos. Por tanto, se impondrá un número fijo de 9 bits para su cuantificación. En cuanto al valor de sobrecarga, se elige en función del valor máximo posible. Por tanto, como la señal de entrada está normalizada, se limita a 1 el máximo valor absoluto de amplitud. Sin embargo, la distribución de las amplitudes de las funciones wavelet-packets dista mucho de ser uniforme; al contrario, se tienen muchas funciones de pequeña amplitud en relación al número de funciones de gran amplitud. Con estas condiciones, el cuantificador usado utiliza una compresión de tipo logarítmico, en concreto ley A, para conseguir en lo posible una relación señal a ruido constante, que depende del número de bits elegido.

8.4.4. Parámetros del ruido

El segmento de señal puede ser dividido en sub-tramas de ruido en base a las decisiones del segmentador de ruido usado. Al igual que en el caso de los segmentos de señal, la información entre sub-tramas de ruido es completamente independiente entre las mismas. Para cada sub-trama de ruido, se codifica la longitud de la sub-trama, la energía total y los parámetros de la envolvente de energía en frecuencia y tiempo. Cada grupo de parámetros se codifica de la siguiente forma:

Tamaño de sub-trama. El tamaño de cada sub-trama de ruido se cuantifica de la misma forma que el tamaño de segmento de señal. Como el tamaño de segmento máximo es de 3072 muestras y el tamaño mínimo de sub-trama de 256 se necesitan 4 bits para codificar este valor.

Energía de ruido. El primer parámetro que se codifica es la energía total de ruido. La cuantificación de este valor se puede realizar bajo criterios psicoacústicos. Así, la máxima energía de ruido posible (la señal está normalizada) corresponde a 96 dB SPL, el valor mínimo al mínimo del umbral de silencio y el número de bits se calcula en función de la máscara de ruido sobre ruido. En este caso, la máscara de ruido sobre ruido elegida es de $26dB$ [Hall98]. El número de bits se calcula como en el caso de las amplitudes de los tonos, según la ecuación 5.35, dando un valor de 7 bits.

Envolvente de ruido en frecuencia. La envolvente de ruido en frecuencia se modela mediante polos *warped*-LPC. En el codificador propuesto el número de polos es variable en función de la ganancia de predicción. Por lo tanto, se codifican dos parámetros, el número de polos y la amplitud de cada polo, de la siguiente forma:

Número de polos warped-LPC. El número de total de polos es función del tamaño de sub-trama de ruido actual, ya que se permite un polo cada 32 muestras de ruido. Sin embargo, se ha establecido un número de bits fijo para este valor independiente del tamaño de sub-trama para evitar que el número de bits dependa de esta información. Para el caso peor, con una sub-trama de ruido de 3072 muestras (el valor máximo) puede haber 96 polos, por lo que se necesitan 7 bits para cuantificar el número de polos.

Amplitud de cada polo warped-LPC. Los coeficientes del filtro se obtienen de la librería *warpTB*. Estos coeficientes de la estructura directa se convierten a coeficientes en celosía. Al ser la señal de entrada una señal real, los coeficientes (en estructura directa o en celosía) son reales, si bien los polos pueden ser reales o complejos conjugados. Los coeficientes de la estructura en celosía tienen valores comprendidos entre 0 y 1. Un valor mayor o igual que 1 indicaría un polo inestable. Si esto ocurre, este polo no se envía. Cuantificando los valores de los coeficientes en celosía, se evitan inestabilidades en el proceso de cuantificación y se obtienen los valores máximo y mínimo del cuantificador de manera directa. Se han empleado 6 bits para la cuantificación de cada coeficiente. En la bibliografía existen otros métodos de cuantificación para los polos, que se pueden revisar en [Kleijn95].

Envolvente de ruido en el tiempo. Básicamente, para la envolvente de ruido en el tiempo se emplean los mismos mecanismos de cuantificación, sólo se deberían variar para la cuantificación de los polos:

Número de polos LPC. Como se permite un polo cada 32 muestras de ruido y, para el caso peor, el tamaño de sub-trama de ruido es de 3072 muestras, puede haber 96 polos, por lo que se necesitan 7 bits para cuantificar el número de polos LPC.

Amplitud de cada polo LPC. El problema de los polos LPC es que al modelarse la transformada del residuo, que es una señal compleja, los polos son también complejos. Para evitar este problema, se modela la transformada del residuo seguido de su extensión simétrica. De esta forma, la señal modelada es par en el tiempo y su transformada real. Este cambio ha de tenerse en cuenta también en el decodificador. Con

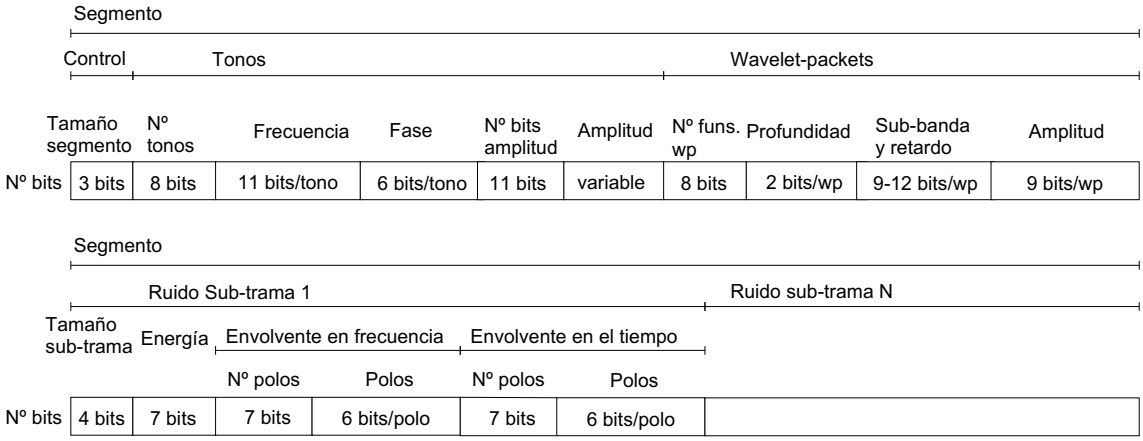

Figura 8.7: *Estructura de la trama binaria del codificador paramétrico propuesto.*

esta solución, los coeficientes de la estructura en celosía son reales y el esquema de cuantificación no cambia.

8.4.5. Estructura de la trama binaria

La estructura de la trama binaria codificada se presenta en la figura 8.7. Se observa como la información de cada segmento es independiente de los demás. Dentro de cada segmento, los datos están separados en datos de control, tonos, funciones wavelet-packets y ruido. A su vez, dentro de cada trama de ruido, la información de cada sub-trama está separada de forma independiente. Es destacable que en cada grupo de información de tonos, transitorios y ruido hay algún parámetro de control para indicar el número de parámetros de cada tipo que se han codificado. El grueso del tamaño de trama final corresponde a los parámetros de los tonos, funciones wavelet-packets y polos de ruido.

Como este codificador está pensado para su uso en una aplicación de *streaming* de audio, es preciso determinar la información más sensible a errores. En este sentido, los parámetros de los que depende la información posterior en la trama son particularmente sensibles, porque un error en ellos provoca la pérdida de sincronismo en el segmento a decodificar. Este hecho es inevitable en un codificador donde el régimen binario es variable cada segmento. Estos parámetros que controlan la información a recibir son:

- Tamaño de segmento.
- Número de tonos.
- Número de bits de las amplitudes de los tonos.
- Número de funciones wavelet-packets.
- tamaño de sub-trama de ruido.
- Número de polos warped-LPC.
- Número de polos LPC.

Así pues, en el diseño de la torre de protocolos donde se envíe la información al decodificador, sería conveniente proteger esta información de manera especial. Además, en lo relativo al diseño de protocolos, al estar la información recibida organizada en segmentos independientes, la información de un segmento es ideal para ser transmitida en paquetes, no teniendo mucho sentido dividir la información de un segmento en paquetes independientes.

Otro tema relacionado con la estructura de la trama del codificador es la cabecera debida a información paramétrica. En un codificador de forma de onda por transformada, sólo es necesario enviar las amplitudes de la transformada. Por ejemplo, si la transformada usada fuera la DFT, la información de amplitud y fase es suficiente, no siendo necesario enviar el índice de frecuencia. Esto se debe a que se envían todas las frecuencias en un codificador por transformada. Para que la compresión en un codificador paramétrico sea satisfactoria, el número de funciones (tonos en el ejemplo) debe ser muy reducido para enviar menos información con el índice incluido (frecuencia en el ejemplo) que en el caso de la codificación por transformada. Por tanto, se conoce como cabecera de información paramétrica los bits dedicados a la definición del átomo dentro del diccionario. Para los tonos, esta cabecera es la frecuencia, mientras que para las funciones wavelet-packets es la profundidad, sub-banda y retardo. En el caso del codificador propuesto, la cabecera gasta más del 50 % del bit rate (en el caso de las funciones wavelet-packets), aunque debido al reducido número de átomos que se modelan el ratio de compresión es elevado. Esta afirmación se comprobará a continuación en los resultados de régimen binario de cada señal de test.

8.5. Resultados

Los resultados del codificador propuesto no se deben presentar de forma aislada, es decir, habrá que comparar los resultados de calidad perceptual y régimen binario con respecto a los resultados obtenidos por otros codificadores comerciales. En concreto, se van a utilizar dos codificadores para realizar la comparación:

1. El codificador AAC con las mejoras introducidas en MPEG-4, porque es el codificador estandarizado de forma de onda que ofrece mayor calidad y de mayor uso en aplicaciones donde el régimen binario es muy reducido.

2. El codificador PPC estandarizado por MPEG [MPEG03], porque es el estándar en codificación paramétrica de audio.

Para comparar los codificadores en igualdad de condiciones, es necesario conocer el régimen binario medio que ofrece el codificador propuesto. En este sentido, para las señales de test utilizadas, el régimen binario resultante es cercano a 16 kbit/s en media. Con este resultado, se comparará con AAC a 16 kbit/s. Sin embargo, para PPC no se ha conseguido el codificador a 16 kbit/s, aunque sí que se han obtenido las señales a 24 kbit/s, gracias a la colaboración del investigador E. G. P. Shuijers de Philips, quien ha aportado las señales codificadas a este régimen binario y un decodificador para obtenerlas. Es preciso tener en cuenta que los regímenes binarios obtenidos no son fijos por segmento, sino que ambos codificadores incluyen estrategias para poder variar de forma instantánea el régimen binario si es recomendable en codificación, aunque el régimen binario objetivo sea el comentado con anterioridad.

Cuadro 8.1: *Régimen binario y otros resultados al codificar el fichero* es01.

Fichero	*es01*
Régimen binario (kbit/s)	17,79
Régimen binario para tonos (kbit/s)	9,02
Régimen binario para wavelet-packets (kbit/s)	1,06
Régimen binario para ruido (kbit/s)	7,63
Régimen binario para AAC-16 (kbit/s)	18,29
Régimen binario para PPC-24 (kbit/s)	24,39
Tamaño medio de segmento (muestras)	1829
Tamaño medio de sub-trama de ruido (muestras)	1125
N de medio de tonos por segmento	17,49
N de medio de funciones wavelet-packets por segmento	2,09
N de medio de polos para la envolvente temporal por sub-trama de ruido	8,76
N de medio de polos para la envolvente frecuencial por sub-trama de ruido	8,20

Las señales de prueba corresponden a las señales propuestas en el CD de la EBU para aseguramiento de la calidad (SQAM) en codificadores de audio. Para cada señal se ha realizado un test de MUSHRA, ya que es la medida subjetiva de la calidad de audio cuando se desean evaluar varios codificadores en la misma prueba. Este test se ha llevado a cabo gracias a la colaboración de 10 miembros del departamento al que pertenezco. Además, para cada señal se presentarán los resultados objetivos del codificador, en especial, el régimen binario que obtiene el codificador propuesto en esta tesis.

8.5.1. Señal *es01*

La señal *es01* es una señal vocal cantada por la artista Suzanne Vega en idioma inglés. Esta señal, incluida en el grupo de las señales vocales, es una señal musical al tratarse de una señal cantada pese a que el único instrumento que aparezca sea la voz. Los resultados objetivos se presentan en la tabla 8.1. Se puede observar como el régimen binario del codificador propuesto es cercano a los 16 kbit/s. Más del 50 % del régimen binario se dedica a información tonal, siendo por otra parte muy reducido el dedicado a transitorios. En cuanto al ruido, el régimen binario es importante, debido a que se usan bastantes polos para parametrizar tanto la envolvente temporal como la frecuencial. El régimen binario dedicado a la cabecera de cada segmento (tamaño de segmento) es muy bajo, concretamente de 0,08 kbit/s. Debido a este bajo régimen binario, para la cabecera de cada trama se obviará esta información en los resultados de régimen binario.

En lo relativo a los resultados subjetivos, el test MUSHRA realizado aparece en la figura 8.8. En esta figura se presenta tanto el valor medio de la calidad valorada por los oyentes como el intervalo del 95 % de confianza de los resultados. El codificador propuesto se ha denominado codificador TWN (*Tones, Wavelet-packets and Noise*) . Los comentarios más reseñables a partir de las opiniones de los oyentes acerca de la calidad de las señales codificadas se presentan a continuación:

- El resultado perceptual de la señal codificada en AAC es el mejor de todos. En esta señal no se escucha otro artefacto que un filtrado con respecto a la señal original.
- Para el codificador PPC aparecen algunos errores propios del modelo, como una señal más

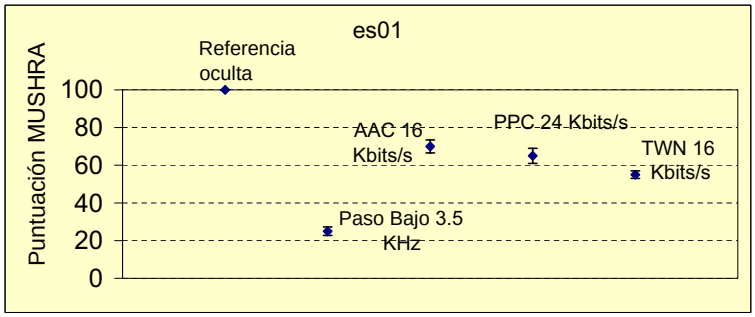

Figura 8.8: *Test MUSHRA para la señal* es01*. Se dibujan los resultados objetivos de la referencia, la señal filtrada paso bajo a 3,5 kHz, la señal codificada con AAC-16 kbit/s, la señal codificada con PPC-24 kbit/s y la señal codificada con el codificador propuesto (TWN). Se representa, tanto el valor medio, como el intervalo del 95 % de confianza para cada versión de la señal evaluada.*

ruidosa que la original. Además, el sonido de los tonos es algo metálico, seguramente debido a que en la codificación de los tonos algunos de ellos se cuantifican con pocos bits (hay un valor fijo de bits por amplitud de cada tono en este codificador).

- El codificador TWN propuesto obtiene el peor resultado perceptual. Esta valoración se atribuye principalmente a que se escucha un eco o desacompañamiento entre las partes tonal y ruidosa de la señal codificada. Es cierto que la energía de la parte ruidosa es aquí mucho mayor que en el caso del codificador PPC, debido a que se extraen muchos menos tonos. Además, es apreciable una señal más ruidosa que en los otros casos.

Tanto en esta como en las demás señales vocales, los resultados de los codificadores paramétricos son peores que para AAC. Esto se debe principalmente a errores del modelo de tonos, transitorios y ruido usado. En una señal vocal y en un fonema sonoro se debería suponer que la parte ruidosa es (casi) nula, y los codificadores paramétricos modelan la energía tonal no audible como ruido lo que provoca una señal ruidosa codificada (hay que tener en cuenta que la máscara de ruido es menor en segmentos tonales que la máscara tonal). Se puede concluir, por tanto, que los codificadores paramétricos de audio no están bien diseñados para codificar señales vocales. Una posible solución es implementar un decisor que indique si la señal es o no vocal para cambiar el codificador en caso de señales vocales.

8.5.2. Señal *es02*

Ésta es una señal vocal masculina hablada en alemán. Para esta señal vocal, los resultados de los codificadores paramétricos no son nada buenos. Por su parte, los resultados objetivos se presentan en la tabla 8.2. Se puede observar cómo el régimen binario del codificador propuesto es un poco superior a los 16 kbit/s. De nuevo, más del 50 % del régimen binario se dedica a información tonal, siendo muy reducido el dedicado a transitorios. Para el ruido, aunque ahora el número de polos es menor, el tamaño de sub-trama de ruido es también menor y el régimen binario se mantiene con respecto al de señal anterior. Salvo en este detalle, los resultados objetivos de ambas señales son muy similares.

El test MUSHRA para esta señal aparece en la figura 8.9. Los resultados subjetivos son

Cuadro 8.2: *Régimen binario y otros resultados al codificar el fichero* es02.

Fichero	*es02*
Régimen binario (kbit/s)	18,44
Régimen binario para tonos (kbit/s)	9,19
Régimen binario para wavelet-packets (kbit/s)	1,29
Régimen binario para ruido (kbit/s)	7,89
Régimen binario para AAC-16 (kbit/s)	19,04
Régimen binario para PPC-24 (kbit/s)	24,74
Tamaño medio de segmento (muestras)	1895
Tamaño medio de sub-trama de ruido (muestras)	915
N de medio de tonos por segmento	18,64
N de medio de funciones wavelet-packets por segmento	2,58
N de medio de polos para la envolvente temporal por sub-trama de ruido	7,24
N de medio de polos para la envolvente frecuencial por sub-trama de ruido	6,65

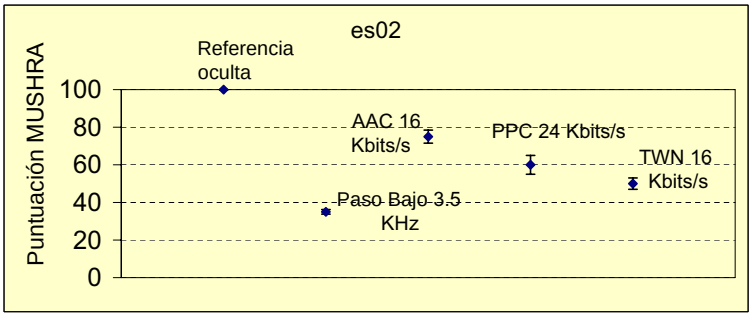

Figura 8.9: *Test MUSHRA para la señal* es02. *Se dibujan los resultados objetivos de la referencia, la señal filtrada paso bajo a 3,5 kHz, la señal codificada con AAC-16 kbit/s, la señal codificada con PPC-24 kbit/s y la señal codificada con el codificador propuesto (TWN).*

también homogéneos en relación a los obtenidos para otras señales vocales. Los comentarios a realizar son los siguientes:

- La señal codificada con AAC obtiene la mejor puntuación en calidad perceptual, apreciándose sólo un ligero filtrado.

- En el caso del codificador PPC, al ser un codificador paramétrico, se nota una señal más ruidosa. De nuevo, se aprecian los tonos sintetizados con carácter metálico.

- Otra vez el codificador TWN propuesto obtiene el peor resultado perceptual. Vuelven a aparecer el eco y una señal ruidosa, aunque ahora si cabe con mayor importancia, ya que las notas perceptuales son menores que en la señal vocal cantada. Una posible explicación es que ahora la señal es menos tonal (al ser hablada y no cantada) en los fonemas sonoros. Se obtiene un residuo con mayor energía y, de aquí, una señal sintética con más deficiencias.

Cuadro 8.3: *Régimen binario y otros resultados al codificar el fichero* es03.

Fichero	*es03*
Régimen binario (kbit/s)	19,38
Régimen binario para tonos (kbit/s)	9,72
Régimen binario para wavelet-packets (kbit/s)	1,33
Régimen binario para ruido (kbit/s)	8,26
Régimen binario para AAC-16 (kbit/s)	19,34
Régimen binario para PPC-24 (kbit/s)	24,72
Tamaño medio de segmento (muestras)	1802
Tamaño medio de sub-trama de ruido (muestras)	868
N de medio de tonos por segmento	18,85
N de medio de funciones wavelet-packets por segmento	2,59
N de medio de polos para la envolvente temporal por sub-trama de ruido	7,10
N de medio de polos para la envolvente frecuencial por sub-trama de ruido	6,67

8.5.3. Señal *es03*

La última de las señales vocales es una señal femenina hablada en inglés. Primero, se presentan los resultados objetivos en la tabla 8.3. En este caso, el régimen binario del codificador propuesto es el mayor de todas las señales vocales. Han crecido en régimen binario todas las componentes de la señal. Este incremento se debe simplemente a que se ha reducido el valor medio, tanto del tamaño de segmento, como del tamaño de sub-trama de ruido.

El test MUSHRA para esta señal se representa en la figura 8.10. Ahora, aparecen nuevos comentarios que realizar en cuanto a los resultados perceptuales de las diferentes señales codificadas:

- Pese a obtener la mejor puntuación, la señal codificada en AAC se aprecia ahora algo más filtrada que en el caso de la señal vocal masculina.
- Para el codificador PPC, la señal se sigue escuchando metálica en los tonos y ruidosa respecto al original.
- Esta vez el codificador TWN ha vuelto a obtener la peor nota. Aparte del eco y la señal ruidosa, ahora se escucha algún pitido de alta frecuencia. Estos pitidos se localizan en segmentos ruidosos, debido al modelado como tonos de alta frecuencia de partes ruidosas de señal. De hecho, algunas *eses* del inglés son casi tonos, pero si se modelan como tales se produce un pitido, que es un artefacto de codificación. Este artefacto es un error del modelo, en concreto, de la extracción tonal.

8.5.4. Señal *si01*

Con esta señal empieza la evaluación de un grupo de tres señales que corresponden a instrumentos en solitario tocando notas aisladas. La señal *si01* se produce con un clavicordio, que es un instrumento de cuerda. La señal está formada por notas aisladas que suben en frecuencia en la escala musical. En cuanto a los resultados objetivos, se presentan en la tabla 8.4. Ahora, el régimen binario del codificador propuesto está más cercano a los 16 kbit/s y se reduce, pese a

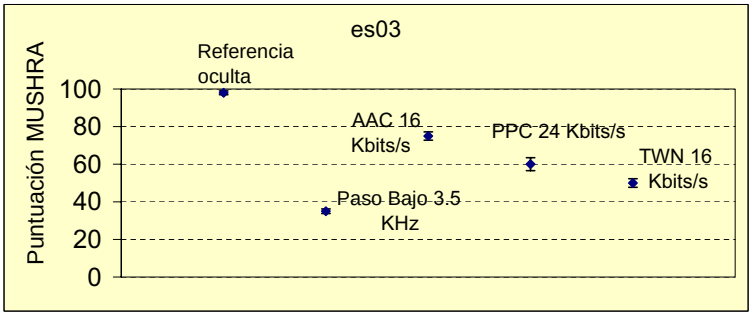

Figura 8.10: *Test MUSHRA para la señal* es03. *Se dibujan los resultados objetivos de la referencia, la señal filtrada paso bajo a 3,5 kHz, la señal codificada con AAC-16 kbit/s, la señal codificada con PPC-24 kbit/s y la señal codificada con el codificador propuesto (TWN).*

Cuadro 8.4: *Régimen binario y otros resultados al codificar el fichero* si01.

Fichero	*si01*
Régimen binario (kbit/s)	17,37
Régimen binario para tonos (kbit/s)	13,19
Régimen binario para wavelet-packets (kbit/s)	0,24
Régimen binario para ruido (kbit/s)	3,85
Régimen binario para AAC-16 (kbit/s)	18,42
Régimen binario para PPC-24 (kbit/s)	22,51
Tamaño medio de segmento (muestras)	1720
Tamaño medio de sub-trama de ruido (muestras)	1164
N de medio de tonos por segmento	24,14
N de medio de funciones wavelet-packets por segmento	0,10
N de medio de polos para la envolvente temporal por sub-trama de ruido	2,09
N de medio de polos para la envolvente frecuencial por sub-trama de ruido	7,11

crecer el régimen binario de los tonos, porque desciende bastante el régimen binario dedicado a la parte ruidosa. Se incrementa el número de frecuencias por segmento en los tonos y disminuyen bastante, tanto el número de polos para la envolvente temporal del ruido, como el número de funciones wavelet-packets por segmento para los transitorios.

Los resultados del test MUSHRA llevado a cabo aparecen dibujados en la figura 8.11. Para esta señal, la primera de las señales musicales evaluadas, la calidad perceptual es muy buena en todos los codificadores evaluados. Se pueden realizar los siguientes comentarios en relación a la calidad perceptual de cada codificador:

- En el codificador AAC sólo se aprecian diferencias con el original debidas a la disminución del ancho de banda de la señal codificada en el comienzo de cada nota.
- El codificador PPC obtienen ahora el peor resultado perceptual. La señal se escucha en general poco natural y el comienzo de cada nota no está bien representado.
- El codificador TWN propuesto obtiene un resultado similar en nota a AAC, aunque los artefactos son bien diferentes. La señal se escucha en este caso algo ruidosa, aunque la representación del comienzo de cada nota es la mejor en este caso.

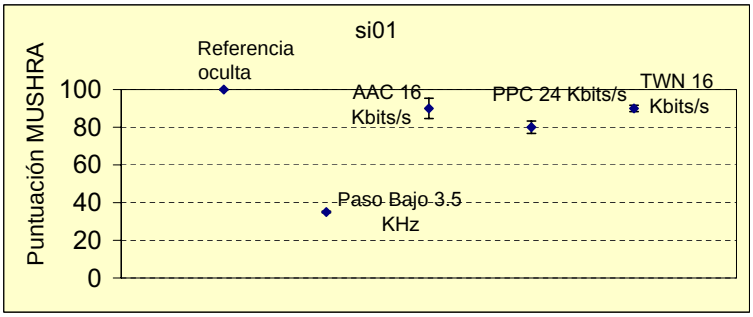

Figura 8.11: *Test MUSHRA para la señal* si01*. Se dibujan los resultados objetivos de la referencia, la señal filtrada paso bajo a 3,5 kHz, la señal codificada con AAC-16 kbit/s, la señal codificada con PPC-24 kbit/s y la señal codificada con el codificador propuesto (TWN).*

Cuadro 8.5: *Régimen binario y otros resultados al codificar el fichero* si02*.*

Fichero	*si02*
Régimen binario (kbit/s)	18,61
Régimen binario para tonos (kbit/s)	6,83
Régimen binario para wavelet-packets (kbit/s)	4,05
Régimen binario para ruido (kbit/s)	7,67
Régimen binario para AAC-16 (kbit/s)	19,05
Régimen binario para PPC-24 (kbit/s)	24,35
Tamaño medio de segmento (muestras)	2571
Tamaño medio de sub-trama de ruido (muestras)	985
N de medio de tonos por segmento	18,84
N de medio de funciones wavelet-packets por segmento	12,12
N de medio de polos para la envolvente temporal por sub-trama de ruido	7,44
N de medio de polos para la envolvente frecuencial por sub-trama de ruido	7,49

8.5.5. Señal *si02*

Esta es la señal con mayor número de transitorios del grupo evaluado. Se trata de una señal producida por una castañuela, por lo que no se puede hablar de notas sino de golpes de castañuela aislados. En los resultados objetivos se aprecia el incremento en régimen binario de la parte transitoria (dedicada a funciones wavelet-packets) con respecto al resto de señales. Aún siendo importante la cantidad de energía modelada por la parte transitoria, el régimen binario que se dedica a esta parte no es elevado. Estos resultados se presentan en la tabla 8.5, donde se observa cómo el régimen binario medio es sólo algo superior a los 16 kbit/s. El régimen binario se mantiene en el orden habitual, debido al aumento del tamaño medio de segmento, puesto que en las zonas con transitorios no se parte la señal en segmentos estacionarios, obteniéndose el tamaño de segmento máximo. Gracias a esta propiedad, el régimen binario de los tonos es muy reducido, aún habiendo un número medio de tonos por segmento similar al de otras señales no transitorias.

Los resultados del test MUSHRA se representan en la figura 8.12. En el caso de esta señal con una importante parte transitoria, la calidad obtenida es muy diversa entre los diferentes codificadores empleados en el test, estando muy relacionada con la calidad obtenida al codificar

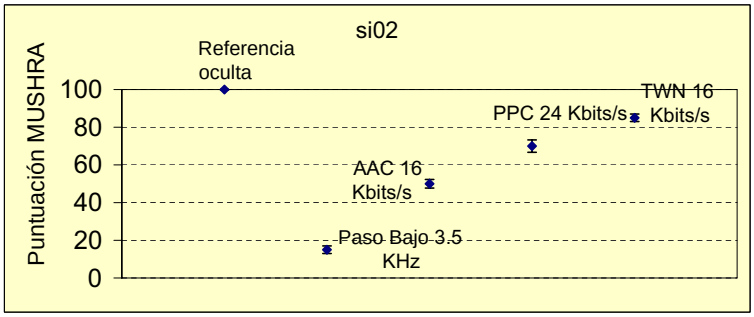

Figura 8.12: *Test MUSHRA para la señal* si02. *Se dibujan los resultados objetivos de la referencia, la señal filtrada paso bajo a 3,5 kHz, la señal codificada con AAC-16 kbit/s, la señal codificada con PPC-24 kbit/s y la señal codificada con el codificador propuesto (TWN).*

los golpes de castañuela. Para cada codificador se puede decir que:

- En el codificador AAC la calidad de los golpes de castañuela es bastante baja. No sólo se filtran al ser codificados, sino que además contienen bastante menos cantidad de energía que los originales, por lo que suenan incluso distorsionados.
- El codificador PPC emplea un modelo de transitorios al parametrizar la envolvente de cada transitorio. Esto funciona correctamente, pero al modelar el contenido espectral de cada transitorio, la parametrización es bastante imprecisa (se supone debido a la limitación en régimen binario). Es conveniente recordar que el codificador PPC normaliza la señal dividiéndola entre su envolvente y realiza entonces un modelado tonal. Como este modelado tonal no se aplica en una señal estacionaria, su resultado desde un punto de vista perceptual es bastante pobre.
- La mejor versión de la señal de castañuelas codificada se obtiene con el codificador TWN propuesto. Ahora, se modela correctamente tanto la energía de cada transitorio como su contenido espectral. Los oyentes observan simplemente un golpe de castañuela algo filtrado respecto del original.

8.5.6. Señal *si03*

Se evalúa ahora una señal muy tonal, ya que está compuesta de notas aisladas producidas por un diapasón. Además, cada una de las tres notas se mantienen en el tiempo varios segundos. En los resultados objetivos que aparecen en la tabla 8.6 lo más destacable es el bajo régimen binario con que se codifica esta señal. Es curioso observar cómo el codificador PPC también obtiene un resultado bajo en régimen binario. La parte tonal domina, por tanto, el régimen binario y se puede ver cómo la envolvente temporal del ruido prácticamente no se utiliza. Además, se observa que el tamaño de segmento es elevado y no se modela ninguna función wavelet-packets en toda la señal, es decir, no se detecta ningún transitorio ni micro-transitorio.

Los resultados del test MUSHRA se representan en la figura 8.13. La señal *si03* es muy tonal, por lo que los artefactos producidos en codificación son básicamente debidos a errores en la parte tonal:

Cuadro 8.6: *Régimen binario y otros resultados al codificar el fichero* si03.

Fichero	*si03*
Régimen binario (kbit/s)	10,70
Régimen binario para tonos (kbit/s)	7,51
Régimen binario para wavelet-packets (kbit/s)	0,17
Régimen binario para ruido (kbit/s)	7,51
Régimen binario para AAC-16 (kbit/s)	17,31
Régimen binario para PPC-24 (kbit/s)	15,26
Tamaño medio de segmento (muestras)	2103
Tamaño medio de sub-trama de ruido (muestras)	1012
N de medio de tonos por segmento	16,00
N de medio de funciones wavelet-packets por segmento	0,00
N de medio de polos para la envolvente temporal por sub-trama de ruido	0,13
N de medio de polos para la envolvente frecuencial por sub-trama de ruido	5,39

- Para el codificador AAC, la calidad es bastante baja porque aparecen artefactos extraños al mantenerse cada nota en el tiempo. El artefacto principal consiste en el efecto *birding*, que en la bibliografía se describe como un efecto molesto que se debe a la respuesta de los bancos de filtros empleados. Aunque este efecto se minimiza en el AAC con las mejoras introducidas por MPEG-4, ocurre en algunas situaciones, como es el caso de esta señal.

- El mejor resultado se obtiene con el codificador PPC, ya que en este caso la parte tonal está muy bien representada. La señal del diapasón tiene una modulación de amplitud en cada nota sostenida, que es bien representada por el diagrama de ventanas que usa el codificador PPC.

- Para el codificador TWN propuesto la señal tiene una calidad aceptable, aunque salen a relucir algunos artefactos. En concreto, el efecto más audible es la aparición/desaparición dentro de la misma nota y de unos segmentos a otros de algunos tonos. Esto produce la sensación de que varía el contenido espectral, cuando la señal en sí es muy estable. La causa a este artefacto se encuentra en que no se realiza un seguimiento entre segmentos de tonos que estén cercanos al umbral de enmascaramiento. Según [Levine98], para evitar este problema, es necesario tener en cuenta para decidir sobre la audibilidad de un tono si dicho tono es también audible en segmentos anteriores. Este proceso de seguimiento lo denomina el autor filtrado (o promediado) de la importancia perceptual.

8.5.7. Señal *sm01*

La primera señal del grupo de tres señales de un sólo instrumento tocando una melodía se trata de la señal producida por una gaita y, por tanto, la señal es bastante tonal. En este instrumento las notas cambian lentamente mientras el instrumento no deja de sonar, es decir, las frecuencias cambian con el tiempo en la frontera entre las notas. En cuanto a los resultados objetivos que se presentan en la tabla 8.7, se pueden destacar varios aspectos. Como en el caso anterior, el régimen binario es inferior a 16 kbit/s, aunque ahora no pasa lo mismo en el codificador PPC. Este bajo régimen binario se debe en gran medida a que el tamaño medio de segmento y, sobre

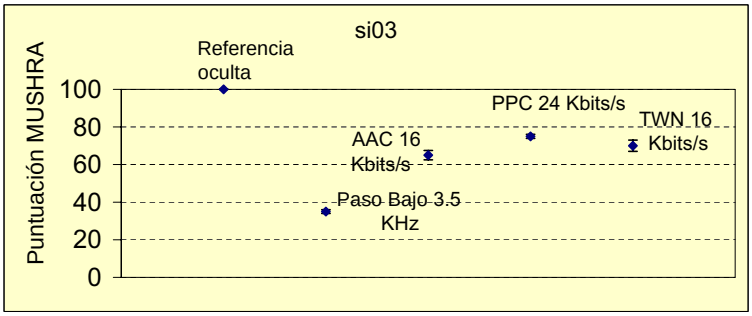

Figura 8.13: *Test MUSHRA para la señal* si03*. Se dibujan los resultados objetivos de la referencia, la señal filtrada paso bajo a 3,5 kHz, la señal codificada con AAC-16 kbit/s, la señal codificada con PPC-24 kbit/s y la señal codificada con el codificador propuesto (TWN).*

Cuadro 8.7: *Régimen binario y otros resultados al codificar el fichero* sm01.

Fichero	*sm01*
Régimen binario (kbit/s)	12,51
Régimen binario para tonos (kbit/s)	9,78
Régimen binario para wavelet-packets (kbit/s)	0,16
Régimen binario para ruido (kbit/s)	2,95
Régimen binario para AAC-16 (kbit/s)	18,34
Régimen binario para PPC-24 (kbit/s)	23,48
Tamaño medio de segmento (muestras)	2257
Tamaño medio de sub-trama de ruido (muestras)	1877
N de medio de tonos por segmento	22,56
N de medio de funciones wavelet-packets por segmento	0,00
N de medio de polos para la envolvente temporal por sub-trama de ruido	0,75
N de medio de polos para la envolvente frecuencial por sub-trama de ruido	9,98

todo, el tamaño de sub-trama de ruido es elevado. Al ser una señal muy tonal, no hay funciones wavelet-packets y se codifican muy pocos polos para modelar la envolvente temporal del ruido.

Los resultados subjetivos derivados del test MUSHRA se encuentran gráficamente en la figura 8.14. Al igual que en la señal anterior, esta señal producida por la gaita es muy tonal y los mayores defectos se aprecian en la parte tonal:

- Se obtiene una señal codificada aceptable para AAC. La mayor distorsión encontrada es el efecto *birding*, aunque no con la intensidad de la señal anterior, por lo que la calificación perceptual final es más alta.
- También es buena la calidad perceptual que obtiene el codificador PPC. Las diferencian respecto al original se manifiestan en una parte tonal con sonido metálico poco realista.
- El codificador TWN propuesto obtiene ahora la nota más baja, al evaluar la calidad subjetiva de la señal de gaita codificada; sin embargo, la calificación obtenida está cercana a la de los otros dos codificadores. Los oyentes manifiestan que la señal codificada se escucha más ruidosa que el original.

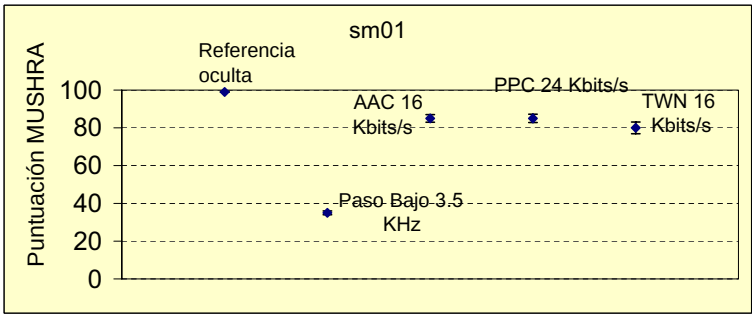

Figura 8.14: *Test MUSHRA para la señal* sm01*. Se dibujan los resultados objetivos de la referencia, la señal filtrada paso bajo a 3,5 kHz, la señal codificada con AAC-16 kbit/s, la señal codificada con PPC-24 kbit/s y la señal codificada con el codificador propuesto (TWN).*

Cuadro 8.8: *Régimen binario y otros resultados al codificar el fichero* sm02.

Fichero	*sm02*
Régimen binario (kbit/s)	6,40
Régimen binario para tonos (kbit/s)	3,15
Régimen binario para wavelet-packets (kbit/s)	0,29
Régimen binario para ruido (kbit/s)	2,88
Régimen binario para AAC-16 (kbit/s)	17,83
Régimen binario para PPC-24 (kbit/s)	12,16
Tamaño medio de segmento (muestras)	2017
Tamaño medio de sub-trama de ruido (muestras)	1184
N de medio de tonos por segmento	6,15
N de medio de funciones wavelet-packets por segmento	0,30
N de medio de polos para la envolvente temporal por sub-trama de ruido	1,13
N de medio de polos para la envolvente frecuencial por sub-trama de ruido	5,93

8.5.8. Señal *sm02*

Esta melodía se produce con el instrumento *glockenspiel*, y es una señal interesante puesto que pone de manifiesto el modelado de los micro-transitorios. Ahora, la señal de una nota no se ha extinguido cuando de repente aparecen las frecuencias de la nueva nota, lo que provoca un repentino aumento de la energía, aunque no con la fuerza suficiente como para hablar de transitorios en la mayoría de los casos. Los resultados de régimen binario junto a otras estadísticas se muestran en la tabla 8.8. Se observa en esta tabla cómo el régimen binario es muy reducido, algo que pasa también en el codificador PPC. La principal causa de este régimen binario se debe a que la parte tonal tiene un muy bajo régimen binario, ya que se modelan pocas frecuencias por segmento en término medio. Pese a esto, el resultado subjetivo es bueno, como se verá a continuación. Es destacable también que aparecen funciones wavelet-packets modeladas, debido a la detección de los micro-transitorios de señal.

Los resultados subjetivos al realizar el test MUSHRA se dibujan en la figura 8.15. En este caso, el modelo de micro-transitorios es fundamental para obtener una buena calidad perceptual:

- La calidad de la señal codificada con AAC es baja. Los errores debidos al efecto *birding* se manifiestan de forma molesta, haciendo que la puntuación obtenida sea menor que en

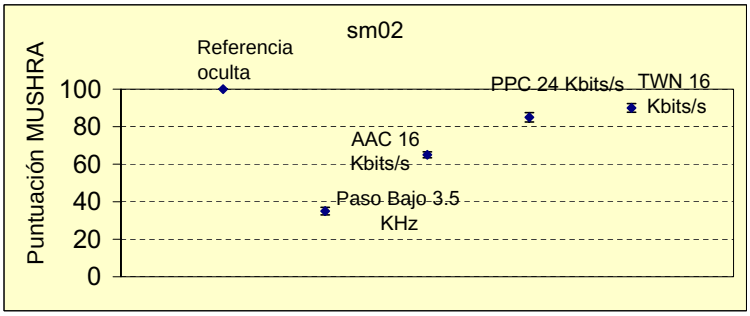

Figura 8.15: *Test MUSHRA para la señal* sm02*. Se dibujan los resultados objetivos de la referencia, la señal filtrada paso bajo a 3,5 kHz, la señal codificada con AAC-16 kbit/s, la señal codificada con PPC-24 kbit/s y la señal codificada con el codificador propuesto (TWN).*

otras ocasiones.

- La calidad obtenida mediante el codificador PPC es aceptable, aunque se noten filtrados los micro-transitorios. Este efecto de filtrado se debe a que el codificador PPC no realiza más que una segmentación adaptativa cuando se encuentra un micro-transitorio. Por lo tanto, los micro-transitorios son modelados con tonos lo que provoca un modelo bastante pobre en lo que a contenido espectral se refiere.

- El mejor modelo de micro-transitorios se obtiene con el codificador TWN propuesto. La señal codificada mantiene una buena riqueza espectral en los micro-transitorios de señal, habiendo sido calificada con la nota más alta por los oyentes.

8.5.9. Señal *sm03*

Esta señal se obtiene mediante punteos de un instrumento de cuerda, en concreto de guitarra. En términos de régimen binario, ver tabla 8.9, esta señal obtiene unos resultados muy cercanos al régimen binario objetivo de 16 kbit/s. Este regimen binario está dominado por los bits dedicados a la parte tonal. El ruido, pese a tener un tamaño medio de sub-trama pequeño, obtiene un régimen binario reducido, ya que casi no se modelan polos para la envolvente temporal. Incluso, aparecen algunas funciones wavelet-packets, ya que se detectan ciertos micro-transitorios en los punteos de guitarra.

En los resultados subjetivos de la figura 8.16, obtenidos mediante el test MUSHRA, se aprecia cómo el codificador TWN propuesto obtiene la mayor calidad perceptual. Se detallan a continuación las causas probables de la puntuación de cada codificador:

- El efecto *birding* vuelve a aparecer en la señal codificada con AAC, lo que hace bajar considerablemente la puntuación subjetiva.

- Si bien la señal codificada con PPC tiene sonidos metálicos por la cuantificación de los tonos, este efecto se aprecia de forma diferente entre oyentes. Como resultado, en media, la calidad obtenida es aceptable.

Cuadro 8.9: *Régimen binario y otros resultados al codificar el fichero* sm03.

Fichero	*sm03*
Régimen binario (kbit/s)	16,90
Régimen binario para tonos (kbit/s)	11,90
Régimen binario para wavelet-packets (kbit/s)	0,22
Régimen binario para ruido (kbit/s)	4,72
Régimen binario para AAC-16 (kbit/s)	18,14
Régimen binario para PPC-24 (kbit/s)	23,99
Tamaño medio de segmento (muestras)	2071
Tamaño medio de sub-trama de ruido (muestras)	899
N de medio de tonos por segmento	25,66
N de medio de funciones wavelet-packets por segmento	0,13
N de medio de polos para la envolvente temporal por sub-trama de ruido	1,07
N de medio de polos para la envolvente frecuencial por sub-trama de ruido	7,83

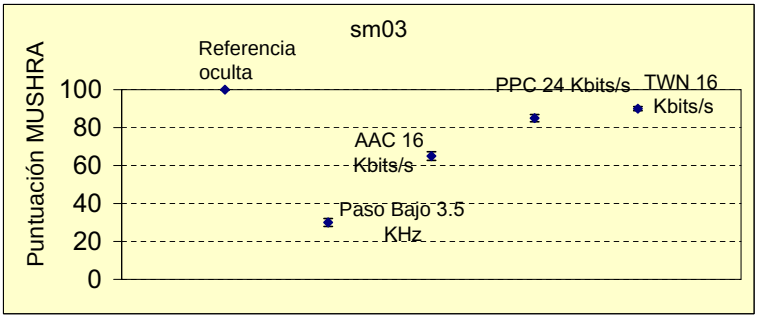

Figura 8.16: *Test MUSHRA para la señal* sm03. *Se dibujan los resultados objetivos de la referencia, la señal filtrada paso bajo a 3,5 kHz, la señal codificada con AAC-16 kbit/s, la señal codificada con PPC-24 kbit/s y la señal codificada con el codificador propuesto (TWN).*

- La mejor puntuación de todos los codificadores se obtiene en esta señal con el codificador TWN propuesto. Los oyentes dan la nota más alta a esta señal codificada, porque obtiene una buena representación de los punteos de guitarra, sin que se aprecie una señal ruidosa como en otras ocasiones.

8.5.10. Señal *sc01*

La primera del grupo de tres señales más complejas es un solo de trompeta. Esta señal se caracteriza por ser una señal tonal con un cambio rápido en el tiempo de las notas musicales. En cuanto a los resultados objetivos, representados en la tabla 8.10, no hay muchas particularidades que poner de manifiesto. La señal obtiene una régimen binario algo superior a los 16 kbit/s. Aunque domina la parte tonal en términos de régimen binario, la parte ruidosa tiene un régimen binario elevado con una gran cantidad de polos tanto para la envolvente espectral como temporal. Este resultado es inquietante, porque indica que la parte ruidosa es importante en energía, aunque se trate de una señal tonal.

Los resultados subjetivos de los tres codificadores evaluados bajo el test MUSHRA en la figura 8.17 son relativamente buenos. Se puede destacar para cada codificador:

Cuadro 8.10: *Régimen binario y otros resultados al codificar el fichero* sc01.

Fichero	*sc01*
Régimen binario (kbit/s)	17,87
Régimen binario para tonos (kbit/s)	10,90
Régimen binario para wavelet-packets (kbit/s)	0,24
Régimen binario para ruido (kbit/s)	6,62
Régimen binario para AAC-16 (kbit/s)	18,65
Régimen binario para PPC-24 (kbit/s)	23,13
Tamaño medio de segmento (muestras)	1844
Tamaño medio de sub-trama de ruido (muestras)	1250
N de medio de tonos por segmento	21,16
N de medio de funciones wavelet-packets por segmento	0,20
N de medio de polos para la envolvente temporal por sub-trama de ruido	8,26
N de medio de polos para la envolvente frecuencial por sub-trama de ruido	8,50

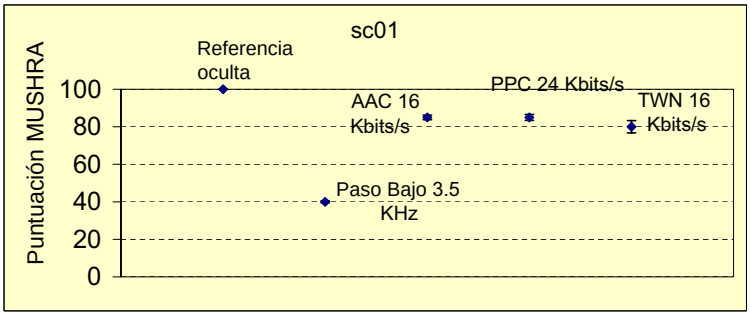

Figura 8.17: *Test MUSHRA para la señal* sc01. *Se dibujan los resultados objetivos de la referencia, la señal filtrada paso bajo a 3,5 kHz, la señal codificada con AAC-16 kbit/s, la señal codificada con PPC-24 kbit/s y la señal codificada con el codificador propuesto (TWN).*

- Con AAC, el comportamiento del proceso de codificación es muy bueno, quizás debido a que se trata de una señal con un contenido espectral concentrado en frecuencias relativamente bajas.
- La calidad de la señal codificada con PPC es también buena, ya que no se aprecian demasiado los sonidos metálicos de la codificación de los tonos en este caso.
- Ahora el codificador PPC obtiene un resultado algo peor en calidad perceptual. La causa se deriva de la fuerte energía de la parte ruidosa, que no es capaz de seguir los rápidos cambios de las notas de la señal original. Este efecto de pre-eco se aprecia en la señal codificada.

8.5.11. Señal *sc02*

La señal *sc02* es una pieza orquestal con unas características bastante tonales. Las notas musicales cambian lentamente con el tiempo. En lo que a régimen binario se refiere, presentado en la tabla 8.11, se encuentra muy cercano a los 16 kbit/s. Como señal tonal, el tamaño de segmento es grande, no hay funciones wavelet-packets y el número de polos de la envolvente temporal del ruido es reducido.

Cuadro 8.11: *Régimen binario y otros resultados al codificar el fichero* sc02.

Fichero	*sc02*
Régimen binario (kbit/s)	16,46
Régimen binario para tonos (kbit/s)	10,60
Régimen binario para wavelet-packets (kbit/s)	0,16
Régimen binario para ruido (kbit/s)	5,63
Régimen binario para AAC-16 (kbit/s)	17,99
Régimen binario para PPC-24 (kbit/s)	24,42
Tamaño medio de segmento (muestras)	2215
Tamaño medio de sub-trama de ruido (muestras)	931
N de medio de tonos por segmento	24,46
N de medio de funciones wavelet-packets por segmento	0,00
N de medio de polos para la envolvente temporal por sub-trama de ruido	2,97
N de medio de polos para la envolvente frecuencial por sub-trama de ruido	7,42

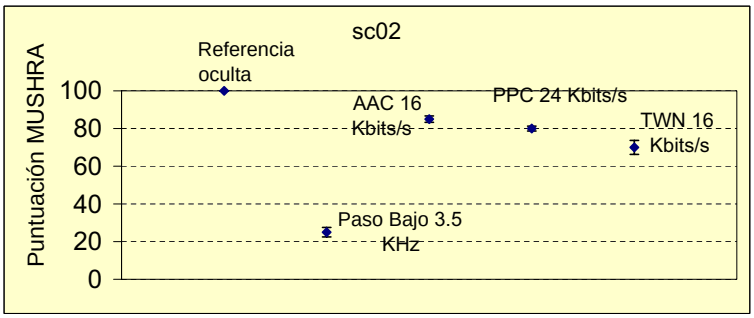

Figura 8.18: *Test MUSHRA para la señal* sc02. *Se dibujan los resultados objetivos de la referencia, la señal filtrada paso bajo a 3,5 kHz, la señal codificada con AAC-16 kbit/s, la señal codificada con PPC-24 kbit/s y la señal codificada con el codificador propuesto (TWN).*

Los resultados subjetivos obtenidos aplicando el test MUSHRA se representan en la figura 8.18. Los comentarios acerca de la calidad de cada señal codificada se realizan a continuación:

- La calidad obtenida con el codificador AAC es muy buena, no apreciándose artefactos importantes.
- Con PPC, la calidad disminuye, debido en gran medida a una parte tonal metálica y a que el ruido es algo diferente del original.
- El codificador TWN obtiene el peor resultado, porque se aprecia un ruido poco natural. La energía modelada con el ruido es mucha y el modelo implementado no obtiene un resultado natural, apreciándose un eco de ruido en la señal codificada.

8.5.12. Señal *sc03*

La última señal evaluada es una señal de pop contemporáneo. Esta señal está formada por varios instrumentos, algunos de ellos de percusión, y tiene una energía cambiante con el tiempo. Los resultados objetivos se muestran en la tabla 8.12. Para esta señal, el codificador propuesto

Cuadro 8.12: *Régimen binario y otros resultados al codificar el fichero* sc03.

Fichero	*sc03*
Régimen binario (kbit/s)	20,76
Régimen binario para tonos (kbit/s)	14,25
Régimen binario para wavelet-packets (kbit/s)	0,23
Régimen binario para ruido (kbit/s)	6,20
Régimen binario para AAC-16 (kbit/s)	18,40
Régimen binario para PPC-24 (kbit/s)	24,77
Tamaño medio de segmento (muestras)	1706
Tamaño medio de sub-trama de ruido (muestras)	1094
N de medio de tonos por segmento	26,07
N de medio de funciones wavelet-packets por segmento	0,05
N de medio de polos para la envolvente temporal por sub-trama de ruido	5,53
N de medio de polos para la envolvente frecuencial por sub-trama de ruido	7,81

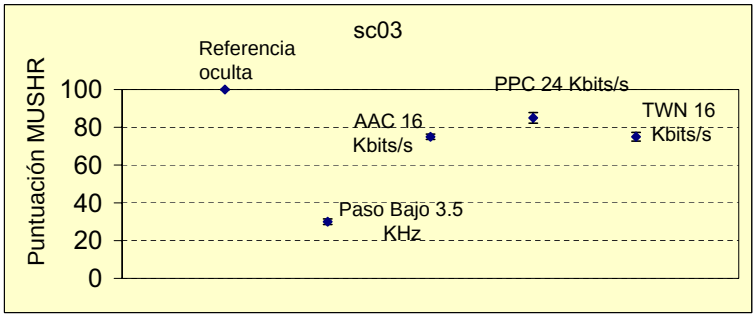

Figura 8.19: *Test MUSHRA para la señal* sc03. *Se dibujan los resultados objetivos de la referencia, la señal filtrada paso bajo a 3,5 kHz, la señal codificada con AAC-16 kbit/s, la señal codificada con PPC-24 kbit/s y la señal codificada con el codificador propuesto (TWN).*

obtiene el mayor régimen binario con un valor algo superior a 20 kbit/s. Se puede observar cómo el régimen binario dedicado a la parte tonal ocupa casi las tres cuartas partes de la asignación de bits. También el ruido requiere un régimen binario alto en relación a otras señales codificadas.

Los resultados subjetivos con el test MUSHRA son similares y aceptables para los tres codificadores evaluados, mostrándose en la figura 8.19. Para cada codificador se puede decir que:

- Con AAC, la señal se escucha algo filtrada en general.
- Con PPC, la señal esta bien representada. Sólo se nota algo metálica en los golpes de percusión.
- Con el codificador TWN propuesto, la señal obtiene una buena representación de los golpes de percusión, aunque se aprecia ruidosa en general.

8.5.13. Resultados en término medio

Por último, se van a mostrar los resultados objetivos y subjetivos en media de todas las señales evaluadas con anterioridad. En lo que a régimen binario se refiere, los resultados en

Cuadro 8.13: *Régimen binario y otros resultados en media al codificar todos las señales evaluadas.*

Todos los ficheros	
Régimen binario (kbit/s)	16,10
Régimen binario para tonos (kbit/s)	9,67
Régimen binario para wavelet-packets (kbit/s)	0,79
Régimen binario para ruido (kbit/s)	5,57
Régimen binario para AAC-16 (kbit/s)	18,40
Régimen binario para PPC-24 (kbit/s)	22,33
Tamaño medio de segmento (muestras)	2003
Tamaño medio de sub-trama de ruido (muestras)	1109
N de medio de tonos por segmento	20,00
N de medio de funciones wavelet-packets por segmento	1,68
N de medio de polos para la envolvente temporal por sub-trama de ruido	4,37
N de medio de polos para la envolvente frecuencial por sub-trama de ruido	7,42

media se muestran en la tabla 8.13. El régimen binario medio está muy próximo a los 16 kbit/s, siendo la mayor parte debido a los tonos. El ruido viene a ocupar un tercio del régimen binario final, mientras que los transitorios tienen un régimen binario medio reducido, ya que aparecen en pocas de las señales tenidas en consideración en esta evaluación. Se puede destacar además que el régimen binario medio de AAC es algo mayor del objetivo marcado de 16 kbit/s.

Los resultados subjetivos medios obtenidos con el test MUSHRA para todas las señales consideradas son similares para los tres codificadores evaluados y se muestran en la figura 8.20. La calidad es ligeramente superior en media para el codificador PPC (que es el que tiene un mayor régimen binario), siendo similar en AAC y el codificador TWN propuesto. El intervalo de confianza es mayor, debido a la variación de las calificaciones entre las distintas señales. Esta variación es mayor en TWN, puesto que se comporta bien para unas señales y relativamente mal para otras, es decir, tiene una alta variabilidad en función del tipo de señal evaluada. Algo similar ocurre en AAC, mientras que en PPC la calificación es más estable entre señales. Para cada codificador se pueden realizar los siguientes comentarios:

- AAC obtiene una muy buena nota en las señales tonales donde no aparece el efecto *birding*; mientras que en aquellas señales tonales donde este efecto se pone de manifiesto la calificación subjetiva obtenida es mucho más baja. En las señales con transitorios y microtransitorios las señales se perciben filtradas y la calidad subjetiva es baja. Sin embargo, en las señales vocales la calidad es elevada.

- PPC es un codificador robusto para las señales evaluadas, obteniendo una calidad aceptable para la mayoría de las señales. Para las señales vocales, PPC, aún teniendo una calidad relativamente baja, se comporta de forma robusta. Las señales con transitorios y microtransitorios tampoco obtienen una buena nota, porque no se obtiene un resultado natural desde un punto de vista perceptual, si bien este codificador ofrece un comportamiento robusto, ya que posee herramientas específicas para modelar estas señales. En las señales tonales, el artefacto más importante es el sonido metálico de la cuantificación de los tonos, en función del cual la calidad puede ser aceptable o muy buena; adicionalmente, algunas señales se escuchan un poco ruidosas. Es importante destacar que, según algunas referencias

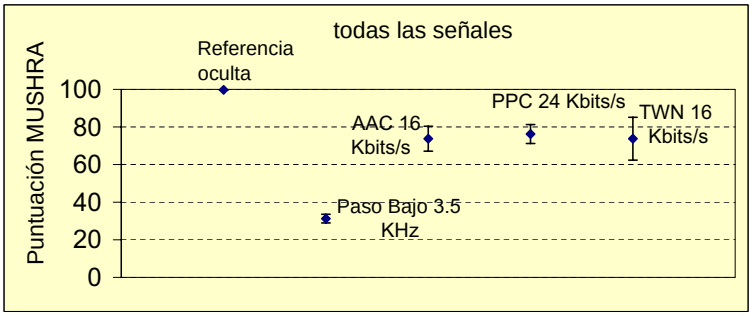

Figura 8.20: *Valores del test MUSHRA en media para todas las señales de prueba. Se dibujan los resultados objetivos de la referencia, las señales filtradas paso bajo a 3,5 kHz, las señales codificadas con AAC-16 kbit/s, las señales codificadas con PPC-24 kbit/s y las señales codificadas con el codificador propuesto (TWN).*

bibliográficas [Brinker02], PPC a 24 kbit/s obtiene una calidad muy similar en media a AAC a 24 kbit/s.

- El codificador TWN propuesto tiene una calidad bastante variable en función de la señal evaluada. Para señales vocales, la calidad es reducida por el ruido poco natural que se escucha como eco, no obteniéndose un resultado satisfactorio. Para señales tonales, la calidad es directamente proporcional a la sensación de señal ruidosa. En ciertas señales tonales este efecto es importante mientras que en otras casi no se aprecia. Los mejores resultados subjetivos se obtienen en señales con transitorios y micro-transitorios, ya que con las herramientas empleadas para su modelado se obtienen unos fantásticos resultados subjetivos manteniendo un régimen binario reducido.

Para terminar este documento, se pasará a realizar los comentarios acerca de las conclusiones que se extraen de este trabajo de investigación y de las importantes líneas futuras que se dibujan como caminos a seguir a partir de los resultados obtenidos.

Parte III

Conclusiones, Líneas Futuras y Publicaciones Generadas

Capítulo 9

Conclusiones

Antes de empezar las conclusiones y líneas futuras de investigación, es preciso tener en cuenta que esta tesis ha estado encaminada al desarrollo de modelos de señal paramétricos con aplicación a la codificación de audio. En este sentido, la aplicación implementada se ha diseñado con orientación a la realización de *streaming* de audio. Además, se ha tenido en cuenta que la adaptación del codificador propuesto hacia una versión escalable sea fácil y sencilla. Con estas consideraciones, en el codificador propuesto la información es completamente independiente entre segmentos de señal. Esta restricción no está presente en el codificador paramétrico estandarizado PPC. En cambio, el codificador AAC sí que tiene implementadas herramientas que permiten que AAC sea escalable y se utilice en algunas aplicaciones de *streaming* de audio.

A partir de los resultados del codificador paramétrico TWN propuesto y de los modelos de señal de transitorios, tonal y de ruido expuestos a lo largo de este documento, se pueden extraer una serie de conclusiones que aclaran la utilidad de las técnicas de codificación implementadas. Las conclusiones extraídas son básicamente las siguientes:

Codificador de audio propuesto. El uso de modelos de señal paramétricos en una aplicación de codificación de audio orientada a realizar *streaming* por internet permite el desarrollo de un codificador completamente paramétrico. Este codificador divide la señal en tonos, transitorios y ruido. Con un modelo tonal con guiado y parada perceptuales, un modelo de transitorios con un diccionario mixto formado por funciones wavelet packets y exponenciales complejas, y un modelo de ruido que obtiene la envolvente en frecuencia mediante *warped*-LPC y la envolvente temporal mediante LPC, es posible diseñar un codificador paramétrico de audio con un régimen binario medio de 16 kbit/s y una buena calidad subjetiva de señal.

Modelo de transitorios. En el codificador paramétrico propuesto, el modelo de transitorios se utiliza para modelizar los transitorios de audio y, también, para obtener un modelo de los micro-transitorios. Para cada caso se aplica en el codificador propuesto de una forma diferente.

- Se puede implementar un modelo paramétrico para los transitorios de señal de audio basado en el empleo del algoritmo *matching pursuits* con un diccionario mixto de funciones wavelet-packets y exponenciales complejas, el cual consigue una calidad excelente en los transitorios codificados con un régimen binario muy reducido. El uso de

este diccionario mixto permite una modelización más exacta de los transitorios de audio con un número de funciones menor que en el caso de los diccionarios en serie. Se ha comprobado, además, como este diccionario mixto obtiene un modelo más apropiado de los transitorios de audio que un diccionario de funciones sinusoidales amortiguadas exponencialmente, que es la aproximación más utilizada en la bibliografía especializada. Para realizar la actualización de las correlaciones en el algoritmo *matching pursuits* con el diccionario mixto propuesto, es necesario tener almacenado en memoria la DFT de las funciones wavelet-packets. Se puede reducir la cantidad de memoria requerida, si se tienen en cuenta las propiedades de desplazamiento en el tiempo de las funciones wavelet-packets que pertenecen al mismo nodo del árbol de descomposición. Se ha llegado a demostrar que se pueden actualizar las correlaciones cruzadas guardando en memoria sólo la FFT de la respuesta de cada nodo del árbol de descomposición WP.

- Para el modelado de los micro-transitorios de audio, un modelo tonal seguido de un modelo de transitorios (con algoritmo *matching pursuits* y un diccionario de funciones wavelet-packets) obtiene un régimen binario muy reducido con una buena representación de estas señales. Se emplea este modelo para los micro-transitorios de audio ante la imposibilidad de detectar esta característica de la señal de audio con un simple detector de transitorios. De hecho, los micro-transitorios sólo pueden ser detectados una vez aplicado el modelo tonal a la señal a analizar. Por tanto, cuando se detecta un micro-transitorio en el residuo que produce el modelo tonal, se aplica un modelo de transitorios basado en el algoritmo *matching pursuits* con un diccionario de funciones wavelet-packets sobre este residuo.

Modelo sinusoidal. El modelo sinusoidal obtiene una señal tonal sintética con una calidad muy natural usando un guiado perceptual en la extracción tonal, una parada del algoritmo *matching pursuits* basada en criterios perceptuales, y un esquema de cuantificación con un número variable de bits por amplitud de cada tono basado en el cálculo de un sencillo umbral de enmascaramiento tanto en codificación como en decodificación. Las contribuciones realizadas en el modelo tonal se detallan a continuación.

- Para realizar el guiado perceptual del algoritmo *matching pursuits* se ha definido una nueva medida de la importancia perceptual de cada tono, basada en la integración en banda de Bark de la división entre la amplitud del tono y el umbral de enmascaramiento. Esta nueva definición en banda de Bark permite una mejor discriminación entre tonos y ruido que la definición en frecuencia encontrada en la bibliografía especializada. Además, la integración en banda de Bark permite una reducción de la complejidad computacional asociada a la medida de importancia perceptual.
- Se puede realizar una parada perceptual del algoritmo *matching pursuits* guiado perceptualmente realizando una correcta inicialización de los umbrales de enmascaramiento. El umbral en la primera iteración del algoritmo *matching pursuits* se debe inicializar al umbral de silencio más el umbral de ruido sobre tonos. La inclusión de este último es necesaria para evitar la selección de tonos que hayan sido enmascarados por el ruido presente en la señal. En cada iteración del algoritmo *matching pursuits* se debe sumar al umbral de enmascaramiento la contribución del tono extraído. De

esta forma, cuando la importancia perceptual para todos los tonos esté por debajo del umbral de enmascaramiento se detiene el algoritmo *matching pursuits*. Con este enfoque, se tienen en cuenta todas las fuentes de enmascaramiento de cada tono y, como consecuencia, se detiene el algoritmo cuando los tonos que quedan en el residuo son inaudibles.

- Para realizar la cuantificación de las amplitudes de cada tono en el modelo sinusoidal, la aproximación más común en la bibliografía especializada es la cuantificación de todas las amplitudes con el mismo número de bits por amplitud y, de esta forma, no tener que enviar información lateral relativa al número de bits por tono. Con este enfoque, la calidad de la señal sintética del modelo tonal se aleja mucho de ser transparente. En esta tesis se ha desarrollado un esquema de cuantificación que permite un número de bits variable para la amplitud de cada tono. Este esquema se basa en el cálculo, tanto en codificación como en decodificación, de un sencillo umbral de enmascaramiento, a partir del cual, estimar el número de bits para la amplitud de cada tono. De esta forma, se evita el envío de información lateral, obteniéndose a la vez una señal sintética con una mejor calidad. En las pruebas realizadas, el esquema de cuantificación propuesto para las amplitudes tonales obtiene mejor calidad y menor régimen binario que otras propuestas encontradas en la bibliografía especializada.

Modelo de ruido. El modelo de ruido parametriza la envolvente en tiempo y frecuencia del residuo dejado por los otros modelos. En el decodificador se genera ruido blanco y se filtra para obtener la misma envolvente en tiempo y frecuencia. En codificación paramétrica de audio, la separación de la señal entre tonos y ruido en el modelo de ruido es un punto crítico. Tanto es así que, en el caso de señales vocales, el modelo que desarrollan los codificadores paramétricos no obtiene una buena representación de estas señales. Para el modelo de ruido se ha trabajado sobre las siguientes cuestiones.

- En relación a la separación entre tonos y ruido, y teniendo en cuenta el modelo tonal propuesto, la señal de entrada se puede dividir teóricamente en tonos audibles, tonos inaudibles y ruido. En este sentido, las simulaciones realizadas no aconsejan eliminar la energía de los tonos inaudibles de señal, sino modelizar esta energía mediante el modelo de ruido. La causa de este resultado se debe a que la separación entre tonos inaudibles y ruido no se puede realizar de forma satisfactoria. El modelado de los tonos inaudibles como ruido, aunque válido en general para señales musicales, no proporciona un buen resultado para las señales vocales, ya que en este caso cuando la señal es sonora no se debe generar una señal ruidosa. Como conclusión, los codificadores paramétricos, basados en la descomposición en un mismo segmento de señal estacionaria de tonos y ruido, se deben utilizar en aplicaciones donde la señal sea una señal musical.
- En base a las pruebas realizadas, se puede considerar en general que un modelo de ruido basado en *warped*-LPC obtiene mejor calidad que un modelo basado en filtros ERB. La explicación de este hecho se encuentra en que el modelo basado en *warped*-LPC obtiene una representación más exacta del residuo en baja frecuencia, que es donde el oído humano tiene una mayor sensibilidad.

- Se puede definir un modelo *warped*-LPC pesado perceptualmente si se modela la envolvente resultante de dividir en frecuencia la señal y el umbral de enmascaramiento. Sin embargo, aunque el modelo de ruido obtenido con este enfoque consigue una señal menos ruidosa, la ventaja introducida no es general para todas las señales, ya que en algunas ocasiones la señal generada por el modelo de ruido se escucha con eco y filtrada. Este resultado inapropiado se obtiene en señales donde el residuo tiene un pitch definido como consecuencia de la importancia de los tonos no audibles. En cualquier caso, el uso de *warped*-LPC pesado perceptualmente se puede considerar una herramienta prometedora en el momento en que se solucionen los problemas de separación entre tonos y ruido.
- Se puede obtener una buena envolvente en tiempo usando la estrategia de TNS sobre el residuo. La idea es utilizar un filtro predictor basado en LPC sobre la transformada del residuo. Al realizar la predicción sobre las muestras de la FFT del residuo, se consigue un filtro LPC en frecuencia cuya respuesta temporal tiene la forma de la envolvente del residuo. Con este esquema se obtiene una envolvente natural, pudiendo determinar el número de polos del filtro predictor a partir de la ganancia de predicción de cada polo. Una vez implementado este esquema en el codificador propuesto se consigue un bajo régimen binario para modelizar la envolvente del residuo en el tiempo.

Capítulo 10

Líneas futuras de investigación

Se dedican los siguientes párrafos a realizar una recopilación de aquellas cuestiones que han quedado abiertas durante la investigación realizada, así como a señalar algunas nuevas direcciones de trabajo en el campo de los modelos de señal paramétricos para señal de audio. Las líneas de investigación que, a juicio del autor, pueden resultar interesantes de abordar en un futuro son las siguientes:

- La primera línea de trabajo abierta a partir del trabajo realizado en esta tesis doctoral es la implementación de una versión escalable del codificador TWN propuesto. En el codificador escalable a desarrollar se debe separar la información de cada segmento en capas, estando formada cada capa por parámetros que tengan una importancia perceptual similar. La idea es partir de una capa básica con un régimen binario reducido que, gracias a la información de las capas superiores, refine la calidad de la señal codificada. Además, para reducir la cantidad de información codificada se debe trabajar en la codificación intra-trama de las frecuencias y las fases del modelo tonal. Una vez realizado este trabajo, la evaluación de este codificador consistiría en la revisión de los protocolos a utilizar en la aplicación de *streaming* por internet que utilice este codificador, en el desarrollo de estrategias de difusión y multidifusión que eviten el corte de la comunicación cuando se producen errores en la transmisión de paquetes o una sobrecarga en la red y, por último, en la evaluación de los codificadores actualmente utilizados en internet para *streaming* de audio. Este trabajo ya se ha comenzado [Cuevas05] [Cuevas06], habiendo sido aprobado en nuestro departamento un proyecto de tesis en esta línea a desarrollar por el profesor Juan Carlos Cuevas Martínez.

- Una línea de trabajo que se puede empezar con independencia de la anterior es la implementación de un codificador paramétrico generalista, que consiga una buena calidad tanto en codificación de música como en codificación de voz. En este sentido, los últimos trabajos realizados [Munoz05] indican que una etapa de pre-procesamiento con extracción de sencillas características a partir de la señal de entrada puede decidir si la señal es música o voz con un alto índice de acierto. En caso de decidir que la señal es musical, el codificador a usar sería, por ejemplo, el codificador propuesto en esta tesis, mientras que si se decide que la señal es de voz, un vocoder parece ser la mejor elección. Con esta decisión música/voz el sistema implementa la primera etapa del árbol de clasificación de señales de audio. De nuevo, los trabajos en esta línea ya se han comenzado con éxito aplicando características

conocidas y un sistema basado en lógica borrosa [Munoz06], y también se ha aprobado en el departamento al que pertenezco un proyecto de tesis a desarrollar por el profesor José Enrique Muñoz Expósito.

- En el campo de los modelos de señal paramétricos con uso en codificación de audio, se pueden comenzar nuevas investigaciones. Así, parece interesante evitar que la extracción tonal guiada perceptualmente dependa de la medida de tonalidad de cada banda. Una posible solución es emplear el modelo de enmascaramiento que aparece en [Par02], donde la máscara se obtiene al pasar la señal de audio por un sistema que imita el tratamiento que se produce en el oído humano. Este sistema devuelve si la señal es o no audible en base a la distorsión producida dentro del sistema. Por otra parte, en los últimos trabajos encontrados en la bibliografía [Gribonval03] se realiza un modelado sinusoidal con extracción de un complejo armónico completo. Esta idea es muy interesante puesto que, por ejemplo, para señal de voz evitaría los errores del modelo, ya que en un fonema sonoro si se extrae todo el complejo armónico generado (en vez de ser tratados como tonos individuales) la separación entre tonos y ruido sería óptima. Se pueden aplicar, en este sentido, las aportaciones realizadas en esta tesis doctoral con respecto al modelado sinusoidal. La idea consiste en tratar al complejo armónico como un conjunto, evitando el tratamiento individualizado de cada tono, que ofrece mucha menos información en la discriminación entre tonos y ruido. Agrupando en la importancia perceptual de cada conjunto armónico la de los tonos individuales que lo forman, se puede conseguir una mejora en la discriminación entre los tonos que pertenecen a un conjunto armónico y aquellos que pertenecen a zonas de ruido, con lo que se mejora sobremanera la distinción entre tonos y ruido, que es el problema más critico encontrado en codificación paramétrica de audio. En cuanto al modelo de ruido, se puede modificar el modelo de ruido para incluir en la generación del mismo, aparte de ruido blanco, la posibilidad de generar ruido multipulso [Ding97]. Con esta opción, se podrá modelar de forma más satisfactoria el residuo cuando éste posee una parte determinística proveniente de tonos no eliminados en la señal.

- Los modelos de señal diseñados a lo largo del trabajo realizado en esta tesis doctoral pueden ser utilizados en otras aplicaciones distintas a la codificación de audio. Un posible campo de aplicación es el desarrollo de sistemas de clasificación de señales de audio en base a los parámetros extraídos por cada modelo. Por ejemplo, las funciones wavelet-packets extraídas en cada señal dan una idea muy precisa del contenido en transitorios y micro-transitorios de la señal en cuestión. Desde otro punto de vista, otra posible aplicación es la separación de fuentes en audio. En este caso, el modelo tonal es muy efectivo al estar basado en consideraciones perceptuales. Por ejemplo, suponiendo una situación práctica, si se desea implementar un sistema que elimine el ruido de fondo en una comunicación móvil realizada dentro de un vehículo, el modelado de ruido contiene la energía correspondiente al ruido de fondo. Así, con un modelo apropiado de los patrones de este ruido, se puede llegar a separar el ruido de fondo de la señal de voz. Incluso, en términos más vagos, puede llegar a ser posible separar dos instrumentos tonales, si, a partir de los tonos audibles, se consigue determinar a qué complejo armónico pertenece cada tono individual, para realizar como consecuencia inmediata la separación de los tonos que pertenecen a cada instrumento.

Capítulo 11

Publicaciones generadas

Revistas incluidas en JCR

1. P. Vera Candeas, N. Ruiz Reyes, D. Martinez Muñoz, M. Rosa Zurera y M. Lucena. Sinusoidal Modelling with Complex Exponentials for Speech and Audio Signals. *Lecture Notes in Computer Science (Springer-Verlag).* Vol. 2652, pp. 1049-1056, Junio, 2003.

2. P. Vera Candeas, N. Ruiz Reyes, M. Rosa Zurera, J. Curpián Alonso y F. López Ferreras. New matching pursuit based sinusoidal modelling method for audio coding. *IEE Proceedings - Vision, Image and Signal Processing.* Vol. 151, pp. 21-28, Febrero, 2004.

3. P. Vera Candeas, N. Ruiz Reyes, M. Rosa Zurera, D. Martinez Muñoz y F. López Ferreras. Transient Modeling by Matching Pursuits with a Wavelet Dictionary for Parametric Audio Coding. *IEEE Signal Processing Letters.* Vol. 11, no. 3, pp 349-352, Marzo, 2004.

4. P. Vera Candeas, N. Ruiz Reyes, M. Rosa Zurera, J.C. Cuevas Martínez y F. López Ferreras. Fast implementation of an improved parametric audio coder based on a mixed dictionary. *Signal Processing.* Vol. 86, no. 3, pp. 432-443, Marzo, 2005.

5. P. Vera Candeas, N. Ruiz Reyes, M. Rosa Zurera, J.C. Cuevas Martínez y F. López Ferreras. Adaptive Signal Models for Wide-Band Speech and Audio Compression. *Lecture Notes in Computer Science (Springer-Verlag).* Vol. 3523, pp. 571-576, Marzo, 2005.

6. P. Vera Candeas, N. Ruiz Reyes, M. Rosa Zurera, J.C. Cuevas Martínez y F. López Ferreras. Sinusoidal Modelling Using Perceptual Matching Pursuits in the Bark Scale for Parametric Audio Coding. *IEE Proceedings - Vision, Image and Signal Processing.* In press, 2006.

Congresos internacionales

1. P. Vera Candeas, N. Ruiz Reyes, M. Rosa Zurera, F. López Ferreras y D. Martínez Muñoz. Matching pursuit based audio coding approach. *2nd Cost Workshops on Information and Knowledge Management for Integrated Media Communication.* Conference proceedings, Florencia, Italia, Marzo, 2002.

2. P. Vera Candeas, N. Ruiz Reyes, M. Rosa Zurera, F. López Ferreras y D. Martínez Muñoz. Energy-adapted matching pursuits in multi-parts models for audio coding purposes. *112th Audio Engineering Society (AES) Convention.* Preprint 5570, Munich, Alemania, Mayo, 2002.

3. P. Vera Candeas, N. Ruiz Reyes, M. Rosa Zurera, D. Martínez Muñoz y M. Lucena. Sinusoidal Modelling with Complex Exponentials for Speech and Audio Signals. *1st Iberian Conference on Pattern Recognition and Image Analysis (IbPRIA 2003).* Conference Proceedings, Palma de Mallorca, España, Junio, 2003.

4. P. Vera Candeas, N. Ruiz Reyes, D. Martinez Muñoz, J. Curpián Alonso, F. Montero de Espinosa y R. Vicen Bueno. High resolution pursuit for detecting flaws close to the surface of strongly scattering materials in NDT applications. *Ultrasonics International 2003.* Granada, España, Julio, 2003.

5. P. Vera Candeas, N. Ruiz Reyes, M. Rosa Zurera y J.M. Fuertes. A new sinusoidal modeling approach for parametric audio coding. *3rd IEEE International Symposium on Image and Signal Processing and Analysis (ISISPA 2003).* Conference Proceedings, Roma, Italia, Septiembre, 2003.

6. P. Vera Candeas, N. Ruiz Reyes, M. Rosa Zurera, J. Curpián Alonso y P.J. Reche López. Signal-adaptive parametric modeling for high quality low bit rate audio coding. *116th AES convention.* Preprint 6176, Berlín, Alemania, Mayo, 2004.

7. P. Vera Candeas, N. Ruiz Reyes, M. Rosa Zurera, J.C. Cuevas Martínez y P.J. Reche López. Parametric audio coding based on adaptive signal models. *12th European Signal Processing Conference (EUSIPCO-2004).* Conference Proceedings, Viena, Austria, Septiembre, 2004.

8. P. Vera Candeas, N. Ruiz Reyes, M. Rosa Zurera, J.C. Cuevas Martínez y J.L. Blanco Claraco. A Sinusoidal Modeling Approach Based on Perceptual Matching Pursuits for Parametric Audio Coding. *118th Audio Engineering Society (AES) Convention.* Convention papers, preprints, Barcelona, España, Mayo, 2005.

9. P. Vera Candeas, N. Ruiz Reyes, M. Rosa Zurera, J.C. Cuevas Martínez y F. López Ferreras. Adaptive Signal Models for Wide-Band Speech and Audio Compression. *2nd Iberian Conference on Pattern Recognition and Image Analysis (IbPRIA 2005).* Conference Proceedings, Estoril, Portugal, Junio, 2005.

10. P. Vera Candeas, N. Ruiz Reyes, M. Rosa Zurera, J.C. Cuevas Martínez y J.M. García. Matching pursuit based on a mixed dictionary composed of Sines + Wavelets for parametric audio coding. *5th EURASIP Conf. on Speech and Image Processing, Multimedia Communications and Services.* Conference Proceedings, Smolenice, Eslovaquia, Julio, 2005.

11. P. Vera Candeas, N. Ruiz Reyes, M. Rosa Zurera, J.C. Cuevas Martínez y J.M. García. Using a Sines + Wavelets mixed dictionary for improving matching pursuit-based parametric audio coding. *13th European Signal processing conference (EUSIPCO-2005).* Conference Proceedings, Antalya, Turquía, Septiembre, 2005.

12. M. Rosa Zurera, P. Vera Candeas, N. Ruiz Reyes y E. Alexandre Cortizo. Parametric audio coding with sparse representations. *11st Symposium AES*. New trends in audio and video. Bialystok, Polonia, Septiembre, 2006.

Otras revistas científicas

1. P. Vera Candeas, N. Ruiz Reyes, D. Martínez Muñoz, J. Curpián Alonso y P.J. Reche López. Post-processing modifications in a parametric audio coder. *WSEAS Transactions on Communications*. Vol. 3, pp. 675-678, Julio, 2004.

Congresos nacionales

1. P. Vera Candeas, M. Rosa Zurera, J. Curpián Alonso y J. Piñeiro. Uso de descomposiciones atómicas para la mejora del modelado sinusoidal en codificación de audio. *XVI Symposium Nacional de la U.R.S.I.*. Actas del congreso, pp. 51-52, Madrid, España, Septiembre, 2001.

Bibliografía

[Adler96] J. Adler, B. Rao, and K. Kreutz-Delgado. Comparison on basis selection methods. *Conference Record 13rd Asilomar Conference on Signals, Systems and Computers*, 1:252–257, November 1996.

[Ali95] M. Ali. *Adaptive signal representation with application in audio coding*. PhD thesis, University of Minnesota, 1995.

[Askenfelt00] A. Askenfelt and A. Galembo. Study of spectral inharmonicity of musical sound by the algorithms of pitch extraction. *Acoustical Physics*, 46(2):121–132, 2000.

[Barbarossa98] S. Barbarossa, A. Scaglione, and G.B. Giannakis. Product high-order ambiguity function for multicomponent polynomial-phase signal modeling. *IEEE Trans. Signal Processing*, 46(3):691–707, 1998.

[Beer92] J.G. Beerends and J.A. Stemerdink. A perceptual audio quality measure based on a psychoacoustic sound representation. *Journal of the AES*, 40(12):963–978, 1992.

[Berg71] T. Berger. *Rate Distortion Theory*. Englewood Cliffs, 1971.

[Brandenburg90] K. Brandenburg and J. Johnston. Second generation perceptual audio coding: The hibrid coder. *Proc. of the 88th AES-Convention*, 1990. Preprint 2937.

[Brandenburg91] K. Brandenburg. Aspec coding. *Proc. of the 10st Int. conf. of the AES*, pages 81–90, 1991.

[Brandenburg97] K. Brandenburg and M. Bosi. Overview of mpeg audio: Current and future standards for low bit-rate audio coding. *Journal of the AES*, 45:4–21, 1997.

[Breebaart04] J. Breebaart, S. van der Par, A. Kohlraush, and E. Schuijers. High-quality parametric spatial audio coding at low bit rates. *Proc. of the 116th AES-Convention*, May 2004. Preprint 6072, Berlin, Germany.

[Brink64] G. ven den Brink. Detection of tone pulse of various durations in noise of various bandwidths. *J. Acoust. Soc. Am.*, 36:1206–1211, 1964.

[Brinker00] A. C. den Brinker, , and A. W. J. Oomen. Fast arma modelling of power espectral density functions. *Proc. of the 10th European Signal Processing Conference (EUSIPCO)*, pages 1229–1232, September 2000. Tampere, Finland.

[Brinker02] A. C. den Brinker, E. G. P. Shuijers, and A. W. J. Oomen. Parametric coding for high quality audio. *Proc. of the 112th AES-Convention*, May 2002. Preprint 5554, Munich, Germany.

[Brinker03] A. C. den Brinker and F Riera-Palou. Pure linear prediction. *Proc. of the 115th AES-Convention*, October 2003. Preprint 5924, New York, USA.

[Brinker95] A. C. den Brinker. Meixner-like functions having a rational z-transform. *Int. J. Circuit Theory Appl.*, 23:237–246, 1995.

[Buus86] S. Buus, E. Schorer, M. Florentine, and E. Zwicker. Decision rules in detection of simple and complex tones. *J. Acoust. Soc. Am.*, 80:1646–1657, 1986.

[Chen95] S.S. Chen. *Basis Pursuit.* PhD thesis, Departament of Statistics. Standford University, 1995.

[Chen95b] S. Chen and J. Wigger. Fast orthogonal least squares algorithm for efficient subset model selection. *IEEE Trans. Signal Processing*, 43(7):1713–1715, 1995.

[Chen96] S.S. Chen, D. Donoho, and M. Saunders. Atomic decomposition by basis pursuit. *Standford University, Tech. Report*, February 1996. Available at playfair.standford.edu.

[Cohen95] L. Cohen. *Time-frequency signal analysis.* Englewood Cliffs, Prentice-Hall, 1995.

[Coifman92] R.R. Coifman and M.V. Wickerhauser. Entropy-based algorithms for best-basis selection. *IEEE Trans. Information Theory*, 38:713–718, 1992.

[Cuevas05] J.C. Cuevas, P. Vera, and N. Ruiz. Scalable parametric audio coder for internet audio streaming. *118th AES convention*, pages Convention papers, preprints, May 2005. Barcelona, Spain.

[Cuevas06] J.C. Cuevas, P. Vera, and N. Ruiz. A community hierarchic based approach for scalable parametric audio multicasting over the internet. *120th AES convention*, pages Convention papers, preprints, May 2006. Paris, France.

[Daubechies88] I. Daubechies. Time-frequency localization operators: a geometric phase space approach. *IEEE Transactions on Information Theory*, 34(4):605–612, 1988.

[Davis94] G. Davis. *Adaptive nonlinear approximations.* PhD thesis, Departament of Mathematics. New York University, 1994.

[Depalle97] Ph. Depalle and T. Hélie. Extraction of spectral peak parameters using a short-time fourier transform modeling and no sidelobe windows. *IEEE ASSP Workshop on Applications of Signal Processing to Audio and Acoustics*, pages 19–22, 1997. New York, USA.

[Desainte00] M. Desainte-Catherine and S. Marchand. High-precision fourier analysis of sounds using signal derivatives. *J. Acoust. Soc. Am.*, 48(7/8):654–667, July/Aug. 2000.

[Dietz02] M. Dietz, L. Liljeryd, K. Kjorling, and O. Kunz. Spectral band replication, a novel approach in audio coding. *Proc. of the 112th AES Convention*, April 2002. Preprint Number 5553.

[Dietz03] M. Dietz. Mpeg-4 extension 1: Bandwidth enhancement. *113th AES Convention*, March 2003. Workshop on Recent Developments in MPEG-4 Audio.

[Ding97] Y. Ding and X. Qian. Sinusoidal and residual decomposition and residual modeling of musical tones using the quasar signal model. *Proc. of the International Computer Music Conference*, pages 35–42, September 1997.

[Edler96] B. Edler, H. Purnhagen, and C. Ferekidis. Asac - analysis/synthesis audio codec for very low bit rates. *Proc. of the 100th AES-Convention*, May 1996. Preprint 4179, Copenhagen, Denmark.

[Edler98] B. Edler and H. Purnhagen. Concepts for hybrid audio coding schemes based on parametric techniques. *Proc. of the 105th AES-Convention*, September 1998. Preprint 5554, San Francisco, USA.

[Fitz95] K. Fitz and L. Haken. Bandwith enhanced sinusoidal modeling in lemur. *Proc. of the International Computer Music Conference*, pages 154–157, September 1995.

[Flanagan66] J.L. Flanagan and R.M. Golden. Phase vocoder. *Bell Laboratories, Tech. Report*, 1966. Bell Syst. Tech. J. 45.

[Fletcher40] H. Fletcher. Auditory patterns. *Rev. Mod. Phys.*, pages 47–65, Enero 1940.

[Fletcher64] H. Fletcher. Normal vibration frequencies of a stiff piano string. *J. Acoust. Soc. Am.*, 36:203–209, 1964.

[Friedlander95] B. Friedlander and A. Zeira. A oversampled gabor representation for transient signals. *IEEE Trans. Signal Processing*, 43:2088–2094, 1995.

[Gabor46] D. Gabor. Theory of communication. *Journal of the Institution of Electrical Engineers*, 93(III-26):429–427, 1946.

[Galembo99] A. Galembo and A. Askenfelt. Signal representation and estimation of spectral parameters by inharmonic comb filters with application to the piano. *IEEE Trans. Speech Audio Processing*, 7(2):197–203, 1999.

[George87] E.B. George and M.J.T. Smith. A new speech coding model based on a least-squares sinusoidal representation. *Proc. IEEE Int. Conf. Acoust. Speech and Signal Proccesing (ICASSP)*, pages 1641–1644, 1987. Dallas, USA.

[George92] E.B. George and M.J.T. Smith. Analysis-by-synthesis/overlap-add sinusoidal modeling applied to the analysis and synthesis of musical tones. *Journal of the Audio Engineering Society*, 40(6):497–515, June 1992.

[George97] E.B. George and M.J.T. Smith. Speech analysis/synthesis and modifications using an analysis-by-synthesis/overlap-add sinusoidal model. *IEEE Trans. on Speech and Audio Processing*, 5(40):389–406, September 1997.

[Gill91] P.E. Gill, W. Murray, and M.H. Wright. *Numerical linear algebra and optimization.* Addison Wesley, 1991. Redwood city, California, USA.

[Glasberg90] B. Glasberg and B.C.J. Moore. Derivation of auditory filter shapes from notched-noise data. *Hearing Res.*, 47:103–138, 1990.

[Gonzalez01] N. Gonzalez and A. Pena. An adaptive tiling of the time-frequency plane with application to multiresolution-based perceptual audio coding. *Signal Processing*, 81:301–319, 2001.

[Goodwin96] Goodwin M. Residual modeling in music analysis/synthesis. *Proc. IEEE Int. Conf. Acoust. Speech and Signal Proccesing (ICASSP)*, pages 1005–1008, Mayo 1996.

[Goodwin97] M.M. Goodwin. *Adaptive signal models: Theory, algorithms, and audio applications.* PhD thesis, Departament of Electrical Engineering and Computer Science at the University of California, 1997.

[Goodwin97b] M.M. Goodwin. Matching pursuit with damped sinusoids. *Proc. IEEE Int. Conf. Acoust. Speech and Signal Proccesing (ICASSP)*, 3:2037–2040, Abril 1997.

[Goodwin98] M.M. Goodwin. *Adaptive signal models: Theory, algorithms, and audio applications.* Kluwer Academic Publishers, 1998.

[Gorodnitsky97] I.F. Gorodnitsky and B.D. Rao. Sparse signal reconstruction from limited data using focuss: A re-weighted minimum norm algorithm. *IEEE Transactions on Signal Processing*, 45(3):600–616, March 1997.

[Greenwood90] D. Greenwood. A cochlear frequency-position function for several species: 29 years later. *J. Acoust. Soc. Amer.*, 87:2592–2605, Junio 1990.

[Gribonval01] R. Gribonval. Fast matching pursuit with a multiscale dictionary o gaussian chirps. *IEEE Trans. on Signal Processing*, 49(5):994–1001, May 2001.

[Gribonval03] R. Gribonval and E. Bacry. Harmonic decompositions of audio signals with matching pursuit. *IEEE Trans. on Signal Processing*, 51(1):101–111, January 2003.

[Gribonval96] R. Gribonval, E. Bacry, S. Mallat, P. Depalle, and X. Rodet. Analysis of sound signals with high resolution matching pursuit. *Proceedings of the IEEE-SP International Symposium on Time-Frequency and Time-Scale Analysis*, pages 125–128, June 1996.

[Griffin88] D.W. Griffin and J.S. Lim. Multiband excitation vocoder. *IEEE Trans. Acoust., Speech, Signal Processing*, 36(8):1223–1235, August 1988.

[Hall98] J. Hall. *Auditory psychophysics for coding applications.* CRC Press, 1998.

[Hamdy96] K.N. Hamdy, M. Ali, and A.H. Tewfik. Low bit rate high quality audio coding with combined harmonic and wavelet representation. *Proc. of ICASSP*, 2:1045–1048, May 1996. Atlanta, Georgia, USA.

[Hamdy99] K.N. Hamdy and A.H. Tewfik. Audio coding using steady state harmonics and residuals. *International Workshop on Multimedia Signal Processing*, September 1999. Copenhagen, Denmark.

[Harma00a] A. Harma, M. Karjalainen, M. Savioja, V. Valimaki, U.K. Laine, and J. Huopaniemi. Frequency warped signal processing for audio applications. *J. Acoust. Eng. Soc.*, 48(11):1011–1031, 2000.

[Harma00b] A. Harma. Implementation of frequency-warped recursive filters. *Signal Processing, Elsevier Science*, 80:543–548, 2000.

[Harma01] A. Harma. *Frequency-warped autoregressive modeling and filtering.* PhD thesis, Helsinki University of Technology. Departament of Electrical and Communications Engineering, 2001.

[Harris78] F.J. Harris. On the use of windows for harmonic analysis with the discrete fourier transform. *Proc. IEEE*, 66(1):51–83, 1978.

[Hawkins50] J.E. Hawkins and S.S. Stevens. The masking of pure tones anf of speech by white noise. *J. Acoust. Soc. Am.*, 22:6–13, 1950.

[Hell72] R.P. Hellman. Asymmetry of masking between noise and tone. *Perception and Psychophysics*, 11(2):241–246, 1972.

[Herre98] J. Herre and D. Schulz. Extending mpeg-4 aac codec by perceptual noise substitution. *Proc. of the 104th AES-Convention*, 1998. Preprint 4720.

[Hess83] W Hess. *Pitch Determination of Speech Signals.* Springer-Verlag, 1983.

[Heusdens00] R. Heusdens and K. Vos. Rate-distortion optimal exponential modeling of audio and speech signals. *Proc. of the 21th Symposium on Information Theory in the Benelux*, pages 77–84, 2000. Wassenaar, The Netherlands.

[Heusdens02] R. Heusdens, R. Vafin, and W.B. Kleijn. Sinusoidal modelling using psychoacoustic-adaptive matching pursuits. *IEEE Signal Proccesing Letters*, 9(4):262–265, 2002.

[Heusdens02b] R. Heusdens and S. van de Par. Rate-distortion optimal sinusoidal modelling of audio and speech using psychoacoustical matching pursuits. *Proc. IEEE Int. Conf. Acoust. Speech and Signal Proccesing (ICASSP)*, II:1809–1812, 2002. Orlando, USA.

[ITU-R01] ITU-R. Method for the subjective assessment of intermediate quality level of coding systems (mushra), 2001. ITU-R Recommend. BS.1534.

[ITU-R01b] ITU-R. Method for objective measurements of perceived audio quality, 2001. ITU-R Recommend. BS.1387-1.

[ITU-R97] ITU-R. Methods for the subjective assessment of small impairments in audio systems including multichannel sound systems, 1997. ITU-R Recommend. BS.1116-1.

[Ikram01] M.Z. Ikram and G.T. Zhou. Estimation of multicomponent polynomial phase signals mixed orders. *Signal Processing*, 81(11):2293–2308, 2001.

[Iwakami95] N. Iwakami, T. Moriya, and S. Miki. High quality audio coding at less than 64 kbit/s using transform-domain interleave vector quantization (twinvq). *Proc. IEEE Int. Conf. Acoust. Speech and Signal Proccesing (ICASSP)*, pages 3095–3098, 1995.

[Jaggi98] S. Jaggi, W.C. Karl, S. Mallat, and A.S. Willsky. High resolution pursuit for feature extraction. *Applied and Computational Harmonic Analysis*, 5(4):428–449, October 1998. Elsevier Science.

[Jensen02] J. Jensen and R. Heusdens. Optimal frequency-differential encoding of sinusoidal model parameters. *Proc. Int. Conf. Acoust. Speech Signal Processing (ICASSP)*, III:2497–2500, 2002. Orlando, USA.

[Johnston88] J.D. Johnston. Estimation of perceptual entropy using noise masking criteria. *Proc. IEEE Int. Conf. Acoust. Speech and Signal Proccesing*, pages 2524–2527, 1988.

[Johnston88b] J.D. Johnston. Transform coding of audio signals using perceptual noise criteria. *IEEE J. Select. Areas. Commun.*, 6(2):314–323, February 1988.

[Kahrs98] M. Kahrs and K. Brandenburg. *Applications of Digital Signal Processing to Audio and Acoustics*. Kluwer Acad-emic Publisher, 1998.

[Kay89] S. Kay. A fast and accurate single frequency estimator. *IEEE Trans. Acoust., Speech, Signal Processing*, 37(12):1987–1999, 1989.

[Kerkhof02] L. van de Kerkhof. Mpeg-4 extension 2: Parametric coding of high-quality audio. *Proc. of the 113th AES-Convention: Workshop on Recent Developments in MPEG-4 Audio*, October 2002.

[Kleijn95] W.B. Kleijn and K.K. Paliwal. *Speech Coding and Synthesis*. Elsevier, 1995.

[Koenen02] R. Koenen. Overview of mpeg-4 standard, March 2002. Technical Report N4030, ISO/IEC JTC1/SC29 WG11.

[Koenen99] R. Koenen. Mpeg-4 overview (maui version), December 1999. Technical Report N3156, ISO/IEC JTC1/SC29 WG11.

[Lam99] Y. Lam and R. Stewart. Perception-based residual analysis-synthesis system. *Proc. IEEE Int. Conf. Acoust. Speech and Signal Proccesing (ICASSP)*, Mayo 1999.

[Langhans92] A. Langhans and A. Kohlrausch. Spectral integration of broadband signals in diotic and dichotic masking experiments. *J. Acoust. Soc. Am.*, 91:317–326, 1992.

[Lattard93] J. Lattard. Influence of inharmonicity on the tunning of a piano - measurements and mathematical simulation. *J. Acoust. Soc. Am.*, 94:46–53, 1993.

[Lee03] P. Lee. *Wavelet Filter Banks in Perceptual Audio Coding.* PhD thesis, University of Waterloo, 2003.

[Lee87] H. Lee, D.P. Sullivan, and T.H. Huang. Improvement of discrete band-limited signal extrapolation by iterative subspace modification. *Proc. IEEE Int. Conf. Acoust. Speech and Signal Proccesing (ICASSP)*, 3:1569–1572, Abril 1987. Dallas, Texas, USA.

[Levine98] S.N. Levine. *Audio representation for data compression and compressed domain processing.* PhD thesis, Departament of Electrical Engineering of Standford University, 1998.

[Levine98b] S.N. Levine, T.S. Verma, and J.O. Smith. Multiresolution sinusoidal modelling for wideband audio with modifications. *Proc. IEEE Int. Conf. on Acoustic, Speech and Signal Processing (ICASSP)*, 6:3585–3588, September 1998. Seattle, USA.

[Levine99] S.N. Levine and J.O. Smith. A switched parametric & transform audio coder. *Proc. IEEE Int. Conf. on Acoustic, Speech and Signal Processing (ICASSP)*, 2:985–988, 1999. Phoenix, USA.

[Liebchen04] T. Liebchen. Mpeg-4 audio lossless coding. *116th AES Convention*, May 2004. Berlin, Germany.

[MPEG01] MPEG. Call for proposals for new tools for audio coding, 2001. ISO/IEC Technical Report JTSC1/SC29/WG11 N3793.

[MPEG03] MPEG. Avc test results validate superior technology, 2001. ISO/IEC Technical Report JTSC1/SC29/WG11 N6085.

[MPEG03b] MPEG. Report on the verification tests of mpeg-4 high efficiency aac, 2003. ISO/IEC Technical Report JTSC1/SC29/WG11 N6009.

[MPEG92] MPEG. Coding of moving pictures and associated audio for digital storage media at up to 1.5mbit/s, part 3: audio, 1992. International Standard IS 11172-3, ISO/IEC JTC1/SC29 WG11.

[MPEG97a] MPEG. Mpeg-2 advanced audio coding, aac, 1997. International Standard IS 13818-7, ISO/IEC JTC1/SC29 WG11.

[MPEG97b] MPEG. Working draft of iso/iec 14496-3 mpeg-4 audio, 1997. Doc. N1631, ISO/IEC JTC1/SC29 WG11.

[MPEG98] MPEG. Information technology - very low bit rate audio-visual coding. part 3: Audio, 1998. International standard 14496-3.

[MPEG99] MPEG. Iso/iec 14496 mpeg-4: Coding of moving pictures and audio, 1999. Doc. N2995, ISO/IEC JTC1/SC29 WG11.

[MPEG99b] MPEG. Iso/iec 14496-3 (mpeg-4 audio): Amd. 1/fpdam, 1999. Doc. N2803, ISO/IEC JTC1/SC29 WG11.

[MPEG99c] MPEG. Iso/iec 14496-3, amd. 1/fpdam: Mpeg-4 audio version 2, 1999. Doc. N3058, ISO/IEC JTC1/SC29 WG11.

[Mallat93] S. Mallat and Z. Zhang. Matching pursuits with time-frequency dictionaries. *IEEE Trans. Signal Proccesing*, 41(12):3397–3415, December 1993.

[Malvar90] H.S. Malvar. Lapped transforms for efficient transform/subband coding. *IEEE Trans. Acoust. Speech and Signal Proccesing*, 38:969–978, 1990.

[Markel76] J. Markel and H. Gray. *Linear Prediction of Speech.* Springer-Verlag, 1976.

[Masri96] P. Masri. *Computer modelling of sound for transformation and synthesis of musical signals.* PhD thesis, University of Bristol, 1996.

[Masri98] P. Masri and N. Canagarajah. Extracting more detail from the scpectrum with phase distortion analysis. *Digital Audio Effects (DAFX) Workshop*, pages 119–122, 1998. Barcelona, Spain.

[Mcaulay86] R.J. McAulay and T.F. Quatieri. Speech analysis/synthesis based on a sinusoidal representation. *IEEE Trans. Acoust., Speech, Signal Proccesing*, 34(4):744–754, 1986.

[Moore83] B. Moore and B. Glasberg. Suggested formulae for calculating auditory-filter bandwidths and excitation patterns. *Journal of the Acoustical Society of America*, 74:750–753, September 1983.

[Moore97] B. Moore. *A introduction to the Psychology of Hearing.* Academic Press, 1997.

[Munoz05] J.E. Munoz, S. Garcia, N. Ruiz, P. Vera, and F. Rivas. Speech/music discrimination using a single warped lpc-based feature. *6th International Conference on Music Information Retrieval (ISMIR 2005)*, page Conference Proceedings, September 2005. London, UK.

[Munoz06] J.E. Munoz, S. Garcia, N. Ruiz, P. Vera, and F. Rivas. A fuzzy rules-based speech/music discrimination approach for intelligent audio coding over the internet. *120th AES convention*, pages Convention papers, preprints, May 2006. Paris, France.

[Myburg01] F.P. Myburg. Sinusoidal analysis of audio with polynomial amplitude and phase, 2001. Philips Research Laboratories, Tech. Rep. Nat.Lab. Technical Note 2001/309.

[Myburg04] F.P. Myburg. *Design of a scalable parametric audio coder.* PhD thesis, Universidad de Eindhoven, 2004.

[Natarajan95] B.K. Natarajan. Sparse approximate solutions to linear systems. *SIAM Journal on Computing*, 24(2):227–234, April 1995.

[Nieuwenhuijse98] J.Ñieuwenhuijse, R. Heusdens, and E.F. Deprettere. Robust exponential modelling of audio signals. *Proc. IEEE Int. Conf. Acoust. Speech and Signal Proccesing (ICASSP)*, VI:3581–3584, 1998. Seattle, USA.

[Painter00] T. Painter and A. Spanias. Perceptual coding of digital audio. *Proceedings of the IEEE*, 88(4):449–513, 2000.

[Painter01] T. Painter and A. Spanias. Perceptual segmentation and component selection in compact sinusoidal representation of audio. *Proc. IEEE Int. Conf. Acoust. Speech and Signal Proccesing (ICASSP)*, 5:3289–3292, May 2001.

[Par02] S. van de Par, A. Kohlraush, G. Charestan, and R. Heusdens. A new psychoacoustical masking model for audio coding applications. *Proc. IEEE Int. Conf. Acoust. Speech and Signal Proccesing (ICASSP)*, II:1805–1808, 2002. Orlando, USA.

[Pati93] Y.C. Pati, R. Rezaiifar, and P.S. Krishnaprasad. Orthogonal matching pursuits: Recursive function approximation with applications to wavelet decomposition. *Proc. 27th Asilomar Conf. Signals, Systems, Computers*, 1993.

[Peeters98] G. Peeters and X. Rodet. Signal characterisation in terms of sinusoidal and non-sinusoidal components. *Proc. Digital Audio Effects*, November 1998. Barcelona, Spain.

[Peleg91] S. Peleg and B. Porat. Estimation and classification of polynomial-phase signals. *IEEE Trans. Information Theory*, 37(2):422–430, 1991.

[Prandoni97] P. Prandoni, M. Goodwin, and M. Vetterli. Optimal time segmentation for signal modeling and compression. *Proc. IEEE Int. Conf. Acoust. Speech and Signal Proccesing (ICASSP)*, pages 2029–2032, 1997. Munich, Germany.

[Princen87] J. Princen, A. Johnson, and A. Bradley. Subband/transform coding using filter bank designs based on time domain aliasing cancelation. *Proc. IEEE Int. Conf. Acoust. Speech and Signal Proccesing (ICASSP)*, pages 2161–2164, 1987.

[Purnhagen00] H. Purnhagen and N. Meine. Hiln - the mpeg-4 parametric audio coding tools. *ISCAS*, III:201–204, May 2000. Geneva, Italy.

[Purnhagen02] H. Purnhagen. Parameter estimation and tracking for time-varying sinusoids. *First IEEE Benelux Workshop on Model based Processing and Coding of Audio (MPCA)*, pages 5–8, November 2002. Leuven, Belgium.

[Purnhagen98] H. Purnhagen, B. Edler, and C. Ferekidis. Object-based analysis/synthesis audio coder for very low bit rates. *Proc. of the 104th AES-Convention*, May 1998. preprint 4747, Amsterdam, The Netherlands.

[Purnhagen99] H. Purnhagen and N. Meine. Core experimental proposal on improved parametric audio coding, March 1999. ISO/IEC Technical Report JTSC1/SC29/WG11 M4492.

[Purnhagen99b] H. Purnhagen. An overview of mpeg-4 audio version 2. *Proc. of the 17th AES International Conference*, pages 157–168, 1999. Florence, Italy.

[Quackenbush03] S. Quackenbush. Mpeg-4 lossless audio coding. *113th AES Convention*, March 2003. Workshop on Recent Developments in MPEG-4 Audio.

[Rabiner78] L.R. Rabiner and R.W. Schafer. *Digital Processing of Speech Signals*. Englewood Cliffs, Prentice-Hall, 1978.

[Rebollo02] L. Rebollo and David Lowe. Optimized orthogonal matching pursuit approach. *IEEE Signal Processing Letters*, 9(4):137–140, April 2002.

[Rodrigues00] J.M. Rodrigues and A.M. Tomé. On the use of backward adaptation in a perceptual audio coder. *IEEE Trans. on Speech and Audio Processing*, 8(4):488–490, July 2000.

[Rosa06] N. Rosa, M. Ruiz, P. Vera, and E. Alexandre. Parametric audio coding with sparse representations. *XI Symposium AES*, September 2006. Bialystok, Poland.

[Rothweiler83] J.H. Rothweiler. Polyphase quadrature filters - a new subband coding technique. *Proc. IEEE Int. Conf. Acoust. Speech and Signal Proccesing (ICASSP)*, pages 1280–1283, 1983.

[Ruiz01] N. Ruiz Reyes. *Codificación de audio basada en la selección de la mejor base de funciones wavelet ortonormales*. PhD thesis, Departamento de Teoría de la Señal y Comunicaciones. Universidad de Alcalá, 2001.

[Ruiz02] N. Ruiz, M. Rosa, F. López, and P. Vera. Algorithm for achieving adaptive tiling of the time axis for audio coding purposes. *IEE Electronics Letters*, 38(9):434–435, 2002.

[Ruiz03] N. Ruiz, M. Rosa, F. López, and P. Jarabo. Adaptive wavelet-packet analysis for audio coding purposes. *Signal Processing, Elsevier Science*, 83(5):919–929, 2003.

[Scharf70] B. Scharf. Critical bands. *Foundations of Modern Auditory Theory*, pages 159–202, 1970.

[Schijndel03] N.H. van Schijndel, M. Gomez, and R. Heusdens. Towards a better balance in sinusoidal plus stochastic representation. *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, pages 197–200, 2003.

[Schijndel99] N.H. van Schijndel, T. Houtgast, and J.M. Festen. Intensity discrimination of gaussian windowed tones: Indications for the shapes of the auditory frequency-time window. *J. Acoust. Soc. Am.*, 105:3425–3435, 1999.

[Schuijers02] E.G.P. Schuijers, A.W.J. Oomen, and A.C. den Brinker. Advances in parametric coding for high-quality audio. *First IEEE Benelux Workshop on Model based Processing and Coding of Audio (MPCA)*, pages 73–79, November 2002. Leuven, Belgium.

[Schuijers03] E. Schuijers. Parametric coding of high-quality audio. *Proc. of the 114th AES-Convention: Workshop on New Technological Developments in MPEG-4 Audio*, March 2003.

[Schuijers04] E.G.P. Schuijers, J. Breebaart, H. Purnhagen, and J. Engdegard. Low complexity parametric stereo coding. *Proc. of the 116th AES-Convention*, May 2004. Preprint 6073, Berlin, Germany.

[Serra89] X. Serra. *A system for sound analysis/transformation/synthesis based on a deterministic plus stochastic decomposition.* PhD thesis, Departament of Music of Standford University, 1989.

[Serra97] X. Serra. *Musical Signal Processing*, chapter Musical sound modelling with sinusoids plus noise. Swets and Zeitlinger, 1997. Curtis Roads, Stephen Pope, Aldo Piccialli and Giovanni De Poli.

[Sluijter99] R.J. Sluijter and A.J.E.M. Janssen. A time warper of speech signals. *Proc. IEEE Workshop on Speech Coding*, pages 150–152, 1999. Porvoo, Finland.

[Smith95] J.O. Smith. The bark bilinear transform. *Proc. of the Workshop on Applications of Single Processing to Audio and Acoustics*, pages 202–205, October 1995.

[Smith99] J.O. Smith III and J.S. Abel. Bark and erb bilinear transforms. *IEEE Transactions on Speech and Audio Processing*, 7:697–708, November 1999.

[Sondhi68] M.M. Sondhi. New methods of pitch extraction. *IEEE Trans. Audio Electroacoust.*, AU-16:262–266, June 1968.

[Soulodre99] G.A. Soulodre and M.C. Lavoie. Subjective evaluation of large and small impairments in audio quality. *AES 17th International Conference on High Quality Audio Coding*, pages 329–336, September 1999. Florence, Italy.

[Stoll00] G. Stoll and F. Kozamernik. Ebu listening tests on internet audio codecs. *EBU Technical Review*, June 2000.

[Strube80] H.W. Strube. Linear prediction on a warped frequency scale. *J. Acoust. Soc. Am.*, 68:1071–1076, 1980.

[Terhardt79] E. Terhardt. Calculation virtual pitch. *Hearing Res.*, 1:155–182, 1979.

[Thomson82] D.J. Thomson. Spectrum estimation and harmonic analysis. *Proceedings of the IEEE*, 70(9):1055–1096, 1982.

[Todd94] C. Todd, G. Davidson, M. Davis, L. Fielder, B. Link, and S. Vernon. Ac-3: Flexible perceptual coding for audio transmission and storage. *96th Audio Engineering Society Convention*, March 1994.

[Tretter85] S.A. Tretter. Estimating the frequency of a noisy sinusoid by linear regresion. *IEEE Trans. Information Theory*, 36(6):832–835, 1985.

[Vera01] P. Vera, M. Rosa, J. Curpian, and J. Piñeiro. Uso de descomposiciones atómicas para la mejora del modelado sinusoidal en codificación de audio. *XVI Symposium Nacional de la U.R.S.I.*, pages 51–52, September 2001. Madrid, Spain.

[Vera02a] P. Vera, N. Ruiz, M. Rosa, F. Lopez, and D. Martinez. Energy-adapted matching pursuits in multi-parts models for audio coding purposes. *112th Audio Engineering Society (AES) Convention*, May 2002. Preprint 5570, Munich, Germany.

[Vera02b] P. Vera, N. Ruiz, M. Rosa, F. Lopez, and D. Martinez. Matching pursuit based audio coding approach. *2nd Cost Workshops on Information and Knowledge Management for Integrated Media Communication*, March 2002. Conference proceedings, Florence, Italy.

[Vera03a] P. Vera-Candeas, N. Ruiz-Reyes, , D. Martinez-Muñoz, J. Curpián-Alonso, F. Montero de Espinosa, and R. Vicen-Bueno. High resolution pursuit for detecting flaws close to the surface of strongly scattering materials in ndt applications. *Ultrasonics International 2003*, July 2003. Conference proceedings, Granada, Spain.

[Vera03b] P. Vera, N. Ruiz, D. Martinez, M. Rosa, and M. Lucena. Sinusoidal modelling with complex exponentials for speech and audio signals. *Lecture Notes in Computer Science (Springer-Verlag)*, 2652:1049–1056, June 2003.

[Vera03c] P. Vera, N. Ruiz, M. Rosa, and J.M. Fuertes. A new sinusoidal modeling approach for parametric audio coding. *3rd IEEE International Symposium on Image and Signal Processing and Analysis (ISISPA 2003)*, September 2003. Conference Proceedings, Roma, Italy.

[Vera03d] P. Vera, N. Ruiz, M. Rosa, , D. Martinez, and M. Lucena. Sinusoidal modelling with complex exponentials for speech and audio signals. *1st Iberian Conference on Pattern Recognition and Image Analysis (IbPRIA 2003)*, June 2003. Conference Proceedings, Palma de Mallorca, Spain.

[Vera04a] P. Vera, N. Ruiz, M. Rosa, D. Martinez, and F. Lopez. Transient modeling by matching pursuits with a wavelet dictionary for parametric audio coding. *IEEE Signal Processing Letters*, 11(3):349–352, Marzo 2004.

[Vera04b] P. Vera, N. Ruiz, M. Rosa, J. Curpián, and F. Lopez. New matching pursuit based sinusoidal modelling method for audio coding. *IEE Proceedings - Vision, Image and Signal Processing*, 151:21–28, Febrero 2004.

[Vera04c] P. Vera, N. Ruiz, M. Rosa, J. Curpián, and P.J. Reche. Signal-adaptive parametric modeling for high quality low bit rate audio coding. *116th AES convention*, May 2004. Preprint 6176, Berlin, Germany.

[Vera04d] P. Vera, N. Ruiz, D. Martinez, J. Curpián, and P.J. Reche. Post-processing modifications in a parametric audio coder. *WSEAS Transactions on Communications*, 3:675–678, July 2004.

[Vera04e] P. Vera, N. Ruiz, M. Rosa, J.C. Cuevas, and P.J. Reche. Parametric audio coding based on adaptive signal models. *12th European Signal processing conference (EUSIPCO-2004)*, September 2004. Conference Proceedings, Vienna, Austria.

[Vera05a] P. Vera, N. Ruiz, M. Rosa, J.C. Cuevas, and F. Lopez. Adaptive signal models for wide-band speech and audio compression. *Lecture Notes in Computer Science (Springer-Verlag)*, 3523:571–576, March 2005.

[Vera05b] P. Vera, N. Ruiz, M. Rosa, J.C. Cuevas, and J.M. Garcia. Using a sines + wavelets mixed dictionary for improving matching pursuit-based parametric audio coding. *13th European Signal processing conference (EUSIPCO-2005)*, September 2005. Conference Proceedings, Antalya, Turkey.

[Vera05c] P. Vera, N. Ruiz, M. Rosa, J.C. Cuevas, and J.M. Garcia. Matching pursuit based on a mixed dictionary composed of sines + wavelets for parametric audio coding. *5th EURASIP Conf. on Speech and Image Processing, Multimedia Communications and Services*, July 2005. Conference Proceedings, Smolenice, Slovakia.

[Vera05d] P. Vera, N. Ruiz, M. Rosa, J.C. Cuevas, and F. López. Adaptive signal models for wide-band speech and audio compression. *2nd Iberian Conference on Pattern Recognition and Image Analysis (IbPRIA 2005)*, June 2005. Conference Proceedings, Estoril, Portugal.

[Vera05e] P. Vera, N. Ruiz, M. Rosa, J.C. Cuevas, and J.L. Blanco. A sinusoidal modeling approach based on perceptual matching pursuits for parametric audio coding. *118th Audio Engineering Society (AES) Convention*, May 2005. Convention papers, preprints, Barcelona, Spain.

[Vera06a] P. Vera, N. Ruiz, M. Rosa, J.C. Cuevas, and F. Lopez. Fast implementation of an improved parametric audio coder based on a mixed dictionary. *Signal Processing*, 86(3):432–443, March 2006.

[Vera06b] P. Vera, N. Ruiz, M. Rosa, J.C. Cuevas, and F. Lopez. Sinusoidal modelling using perceptual matching pursuits in the bark scale for parametric audio coding. *IEE Proceedings - Vision, Image and Signal Processing*, In press, 2006.

[Verma00] T.S. Verma and T.H.Y. Meng. A 6 kbps to 85 kbps scalable audio coder. *Proc. Int. Conf. Acoust. Speech Signal Processing (ICASSP)*, pages 877–880, 2000. Istanbul, Turkey.

[Verma98] T.S. Verma and T.H.Y. Meng. An analysis/synthesis tool for transient signals that allows a flexible sines+transients+noise model for audio. *Proc. Int. Conf. Acoust. Speech Signal Processing (ICASSP)*, pages 12–15, May 1998. Seattle, WA, USA.

[Verma99] T.S. Verma. *A perceptually based audio signal model with application to scalable audio compression.* PhD thesis, Departament of Electrical Engineering of Standford University, 1999.

[Verma99b] T.S. Verma and T.H.Y. Meng. Sinusoidal modeling using frame-based perceptually weighted matching pursuits. *Proc. Int. Conf. Acoust. Speech Signal Processing (ICASSP)*, pages 981–984, 1999.

[Vos99] K. Vos, R. Vafin, R. Heusdens, and W.B. Kleijn. High-quality consistent analysis-synthesis in sinusoidal coding. *AES 17th International Conference on High Quality Audio Coding*, pages 244–250, September 1999. Florence, Italy.

[Waters98] G.T. (Editor) Waters. Sound quality assessment material recordings for subjective tests. users' handbook for the ebu - sqam compact disc, April 1998. Technical centre of the European Broadcasting Union, Tech. Rep. 3253-E.

[Xiong97] Z. Xiong, K. Ramchandran, C. Herley, and M.T. Orchard. Flexible tree-structured signal expansions using time-varying wavelet packets. *IEEE Trans. Signal Processing*, 45(2):333–345, February 1997.

[Yost85] W.A. Yost and D.W. Nielsen. *Fundamentals of hearing. An introduction.* Holdt, Rinehart and Winston, 1985.

[Ziegler02] T. Ziegler, A. Ehret, and M. Ekstrand, P. Lutzky. Enhancing mp3 with sbr: Features and capabilities of the new mp3pro algorithm. *Proc. of the 112th AES Convention*, April 2002. Preprint Number 5560.

[Zwicker82] E. Zwicker and A. Jaroszewski. Inverse frequency dependance of simultaneous tone-on-tone masking patterns at low levels. *J. Acoust. Soc. Am.*, 71:1508–1512, 1982.

[Zwicker90] E. Zwicker and H. Fastl. *Psychoacoustics, Facts and Models.* Springer, 1990.

[Zwicker99] E. Zwicker and H. Fastl. *Psychoacoustics, Facts and Models, 2nd Edition.* Springer, 1999.

Índice alfabético