

19



OFICINA ESPAÑOLA DE  
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 378 393**

51 Int. Cl.:  
**H03M 7/30** (2006.01)  
**H04N 7/30** (2006.01)  
**G10L 19/00** (2006.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

- 96 Número de solicitud europea: **06787173 .1**  
96 Fecha de presentación: **14.07.2006**  
97 Número de publicación de la solicitud: **1905000**  
97 Fecha de publicación de la solicitud: **02.04.2008**

54 Título: **Utilización selectiva de múltiples modelos para codificación y decodificación adaptativa**

30 Prioridad:  
**15.07.2005 US 183266**

45 Fecha de publicación de la mención BOPI:  
**12.04.2012**

45 Fecha de la publicación del folleto de la patente:  
**12.04.2012**

73 Titular/es:  
**MICROSOFT CORPORATION  
ONE MICROSOFT WAY  
REDMOND, WASHINGTON 98052-6399, US**

72 Inventor/es:  
**CHEN, Wei-Ge y  
MEHROTRA, Sanjeev**

74 Agente/Representante:  
**Carvajal y Urquijo, Isabel**

ES 2 378 393 T3

Aviso: En el plazo de nueve meses a contar desde la fecha de publicación en el Boletín europeo de patentes, de la mención de concesión de la patente europea, cualquier persona podrá oponerse ante la Oficina Europea de Patentes a la patente concedida. La oposición deberá formularse por escrito y estar motivada; sólo se considerará como formulada una vez que se haya realizado el pago de la tasa de oposición (art. 99.1 del Convenio sobre concesión de Patentes Europeas).

**DESCRIPCIÓN**

Utilización selectiva de múltiples modelos entrópicos para codificación y descodificación adaptativa

**ANTECEDENTES**

5 Los ingenieros utilizan diversas técnicas para procesar eficientemente el audio digital, manteniendo a la vez la calidad del audio digital. Para comprender estas técnicas, resulta de ayuda comprender cómo la información de audio es representada y procesada en un ordenador.

I. Representación de información de audio en un ordenador.

10 Un ordenador procesa la información de audio como una serie de números que representan la información de audio. Por ejemplo, un único número puede representar una muestra de audio, que es un valor de amplitud en un tiempo dado. Diversos factores afectan a la calidad de la información de audio, incluyendo la profundidad de la muestra, la velocidad de muestreo y el modo de canal.

15 La profundidad (o precisión) de la muestra indica el rango de números utilizados para representar una muestra. Cuanto más valores posibles hay para la muestra mayor es la calidad, debido a que el número puede capturar variaciones de amplitud más sutiles. Por ejemplo, una muestra de 8 bits tiene 256 posibles valores, mientras que una muestra de 16 bits tiene 65 536 posibles valores.

La velocidad de muestreo (habitualmente medida como el número de muestras por segundo) afecta asimismo a la calidad. A mayor velocidad de muestreo, mayor calidad debido a que pueden representarse más frecuencias de sonido. Algunas velocidades de muestreo comunes son 8000, 11 025, 22 050, 32 000, 44 100, 48 000, y 96 000 muestras/segundo.

20 Dos modos de canal comunes para el audio son mono y estéreo. En modo mono, la información de audio está presente en un canal. En modo estéreo, la información de audio está presente en dos canales, denominados normalmente canales izquierdo y derecho. Asimismo, son posibles otros modos con más canales, tales como el sonido envolvente del canal 5.1, el canal 7.1, o el canal 9.1 (el "1" indica un canal de efectos de graves o de baja frecuencia). La tabla 1 muestra varios formatos de audio con diferentes niveles de calidad, junto con correspondientes costos en velocidad de bits en bruto.

25

Tabla 1. Velocidades de bits para información de audio de diferente calidad.

	Profundidad de la muestra (bits/muestra)	Velocidad de muestreo (muestras/segundo)	Modo del canal	Velocidad de bits en bruto (bits/segundo)
Telefonía por internet	8	8	mono	64
Teléfono	8	11,025	mono	88,2
Aucio CD	16	44,1	estéreo	1 411 200

30 El audio de sonido envolvente tiene habitualmente una velocidad de bits en bruto incluso mayor. Tal como muestra la tabla 1, el costo de una información de audio de alta calidad es una velocidad de bits elevada. La información de audio de alta calidad consume grandes cantidades de almacenamiento informático y de capacidad de transmisión. Sin embargo, las compañías y los consumidores dependen cada vez más de los ordenadores para crear, distribuir y reproducir contenido de audio de alta calidad.

II. Procesamiento de información de audio en un ordenador.

35 Muchos ordenadores y redes informáticas carecen de recursos para procesar audio digital en bruto. La compresión (denominada, asimismo, codificación o descodificación) reduce el costo de almacenar y transmitir información de audio, transformando la información a una forma con una menor velocidad de bits. La compresión puede ser sin pérdidas (en la cual la calidad no es afectada) o con pérdidas (en la cual la calidad es afectada, pero la reducción de la velocidad de bits a partir de la subsiguiente compresión sin pérdidas es más dramática). Por ejemplo, la comprensión con pérdidas se utiliza para aproximar información de audio original, y a continuación la aproximación se comprime sin pérdidas. La descompresión (denominada, asimismo, descodificación) extrae una versión reconstruida de la información original, a partir de la forma comprimida.

40

Un objetivo de la compresión de audio es representar digitalmente señales de audio para proporcionar la máxima calidad de señal percibida, con la menor cantidad bits posible. Con este objetivo como meta, diversos sistemas presentes de codificación de audio utilizan modelos de la percepción humana. Sistemas de codificador y descodificador incluyen ciertas versiones del codificador y descodificador Windows Media Audio ("WMA") y del codificador y descodificador WMA Pro, de Microsoft Corporation. Otros sistemas están especificados por ciertas versiones del estándar Audio Layer 3 (capa de audio 3) del Motion Picture Experts Group (grupo de expertos en imágenes en movimiento), el estándar Advanced Audio Coding (codificación de audio avanzada) ("AAC"), y Dolby AC3. Dichos sistemas utilizan habitualmente una combinación de compresión y descompresión con pérdidas y sin pérdidas.

5

10 A. Compresión con pérdidas y descompresión correspondiente.

Convencionalmente, un codificador de audio utiliza una serie de diferentes técnicas de compresión con pérdidas. Estas técnicas de compresión con pérdidas involucran habitualmente modelización/ponderación perceptual y cuantificación después de una transformada de frecuencia. La correspondiente descompresión involucra cuantificación inversa, ponderación inversa y transformadas de frecuencia inversas.

15

Las técnicas de transformadas de frecuencia transforman los datos a una forma que facilita separar la información importante perceptualmente respecto de la información no importante perceptualmente. A continuación, la información menos importante puede ser sometida a una compresión con más pérdidas, mientras que la información más importante es preservada, con objeto de proporcionar la mejor calidad percibida para una velocidad de bits dada. Habitualmente, una transformada de frecuencia recibe muestras de audio y las convierte en datos en el dominio de frecuencias, en ocasiones denominados coeficientes de frecuencia o coeficientes espectrales.

20

La modelización perceptual involucra el procesamiento de datos de audio de acuerdo con un modelo del sistema auditivo humano, para mejorar la calidad percibida de la señal de audio reconstruida para una velocidad de bits dada. Utilizando los resultados de la modelización perceptual, un codificador conforma ruido (por ejemplo, el ruido de cuantificación) en los datos de audio, con el objetivo de minimizar la audibilidad del ruido para una velocidad de bits dada.

25

La cuantificación mapea rangos de valores de entrada a valores únicos, introduciendo pérdidas de información irreversibles pero, asimismo, permitiendo a un codificador regular la calidad y la velocidad de bits de la salida. En ocasiones, el codificador lleva a cabo la cuantificación junto con un controlador de velocidad que ajusta la cuantificación para regular la velocidad de bits y/o la calidad. Existen varias clases de cuantificación, incluyendo adaptativa y no adaptativa, escalar y vectorial, y uniforme y no uniforme. La ponderación perceptual puede considerarse una forma de cuantificación no uniforme.

30

La cuantificación inversa y la ponderación inversa reconstruyen los datos de coeficientes de frecuencia ponderados, cuantificados, a una aproximación de los datos de coeficientes de frecuencia originales. A continuación, una transformada de frecuencias inversa convierte los datos de coeficientes de frecuencia reconstruidos, en muestras de audio reconstruidas en el dominio temporal.

35

B. Compresión y descompresión sin pérdidas.

Convencionalmente, un codificador de audio utiliza una o varias de una serie de diferentes técnicas de compresión sin pérdidas, que se denominan asimismo técnicas de codificación entrópicas. En general, las técnicas de compresión sin pérdidas incluyen codificación por longitud de serie, codificación de longitud variable, y codificación aritmética. Las correspondientes técnicas de descompresión (denominadas, asimismo, técnicas de descompresión entrópicas) incluyen descodificación por longitud de serie, descodificación de longitud variable, y descodificación aritmética.

40

La codificación de longitud de serie es una técnica de compresión simple, bien conocida. En general, la codificación de longitud de serie sustituye una secuencia (es decir, una serie) de símbolos consecutivos que tienen el mismo valor, por el valor y la longitud de la secuencia. En la descodificación por longitud de serie, la secuencia de símbolos consecutivos es reconstruida a partir del valor de la serie y la longitud de la serie. Se han desarrollado numerosas variaciones de codificación/descodificación por longitud de serie.

45

Se proporciona un ejemplo de una variación de este tipo en el documento US2005/0015249, publicado el 20 de enero de 2005, y que describe un codificador de audio que conmuta entre codificación Huffman vectorial de dimensión variable, de niveles directos de datos de audio cuantificados, y codificación de serie-nivel de longitudes de serie y niveles de datos de audio cuantificados.

50

La codificación de serie-nivel es similar a la codificación por longitudes de serie, en cuanto que las series de símbolos consecutivos que tienen el mismo valor son sustituidas por las longitudes de las series. El valor para las

series es el valor predominante (por ejemplo, 0) en los datos, y las series son separadas mediante uno o varios niveles que tienen un valor diferente (por ejemplo, un valor distinto de cero).

5 Los resultados de la codificación por longitud de serie (por ejemplo, los valores de la serie y las longitudes de la serie) o de la codificación de serie-nivel, pueden ser sometidos a codificación de longitud variable para reducir más la velocidad de bits. En este caso, los datos sometidos a codificación de longitud variable son sometidos a descodificación de longitud variable antes de la descodificación por longitud de serie.

10 La codificación de longitud variable es otra técnica de compresión bien conocida. En general, una tabla de códigos de longitud variable ["VLC" (variable length code)] asocia VLCs con valores de símbolo únicos (o combinaciones únicas de valores). Los códigos Huffman son un tipo común de VLC. Se asignan códigos más cortos a valores de símbolo más probables, y se asignan códigos más largos a valores de símbolo menos probables. Las probabilidades son calculadas para ejemplos típicos de alguna clase de contenido. O bien, las probabilidades son calculadas para datos recién codificados o datos a codificar, en cuyo caso los VLC se adaptan a probabilidades cambiantes para los valores de símbolo únicos. En comparación con la codificación estática de longitud variable, la configuración adaptativa de longitud variable reduce habitualmente la velocidad de bits de los datos comprimidos, al incorporar probabilidades más precisas para los datos, pero es necesario asimismo transmitir información extra que especifica los VLCs.

Para codificar símbolos, un codificador de longitud variable sustituye valores de símbolo con los VLC asociados con los valores de símbolo en la tabla de VLC. Para descodificar, un descodificador de longitud variable sustituye los VLC por los valores de símbolo asociados con los VLC.

20 En codificación escalar de longitud variable, una tabla VLC asocia un único VLC a un valor, por ejemplo, un nivel directo de un valor de datos cuantificados. En codificación vectorial de longitud variable, una tabla VLC asocia un único VLC a una combinación de valores, por ejemplo, un grupo de niveles directos de valores de datos cuantificados en un orden particular. La codificación vectorial de longitud variable puede conducir a una reducción de la velocidad de bits mejor que la codificación escalar de longitud variable (por ejemplo, permitiendo al codificador explotar probabilidades fraccionalmente, en VLCs binarios). Por otra parte, la tabla de VLCs para codificación vectorial de longitud variable puede ser extremadamente grande, cuando códigos únicos representan grandes grupos de símbolos, o símbolos que tienen grandes rangos de valores potenciales (debido al gran número de combinaciones potenciales), lo que consume memoria y recursos de procesamiento para calcular la tabla VLC y para encontrar los VLCs. Se han desarrollado numerosas variaciones de codificación/descodificación de longitud variable.

35 La codificación aritmética es otra técnica de compresión bien conocida. La codificación aritmética se utiliza en ocasiones en aplicaciones en las que el número óptimo de bits para codificar un símbolo de entrada dado es un número de bits fraccionario, y en casos en los que existe una correlación estadística entre ciertos símbolos de entrada individuales. En general, la codificación aritmética implica representar una secuencia de entrada como un único número dentro de un rango dado. Habitualmente, el número es un número fraccionario entre 0 y 1. Los símbolos en la secuencia de entrada están asociados con rangos que ocupan partes del espacio entre 0 y 1. Los rangos se calculan en base a la probabilidad de que se produzca el símbolo concreto en la secuencia de entrada. El número fraccionario utilizado para representar la secuencia de entrada es construido haciendo referencia a los rangos. Por lo tanto, las distribuciones de probabilidad para los símbolos de entrada son importantes en los esquemas de codificación aritmética.

45 En la codificación aritmética basada en contexto, diferentes distribuciones de probabilidad para los símbolos de entrada están asociadas con contextos diferentes. La distribución de probabilidad utilizada para codificar la secuencia de entrada cambia cuando el contexto cambia. El contexto puede ser calculado midiendo diferentes factores que se supone afectan a la probabilidad de que aparezca un símbolo de entrada concreto en una secuencia de entrada.

Dada la importancia de la compresión y la descompresión en el procesamiento de medios, no es sorprendente que la compresión y la descompresión sean campos muy desarrollados. Cualesquiera sean las ventajas de las técnicas y los sistemas previos para compresión y descompresión sin pérdidas, no obstante, carecen de diversas ventajas de las técnicas y los sistemas descritos en el presente documento.

## 50 RESUMEN

La invención está definida mediante las reivindicaciones independientes 1 y 16, respectivamente.

En el presente documento se describen técnicas y herramientas para utilizar selectivamente múltiples modelos entrópicos en codificación y descodificación adaptativas. Por ejemplo, utilizar selectivamente múltiples modelos entrópicos puede reducir significativamente la utilización de recursos para múltiples distribuciones/tablas VLC. Al

mismo tiempo, puede conseguirse gran parte de la ganancia de codificación asociada con la utilización de múltiples distribuciones/tablas VLC.

5 De acuerdo con un primer conjunto de técnicas y herramientas, una herramienta tal como un codificador o un descodificador, para símbolos, selecciona un modelo entrópico a partir de un primer conjunto de modelos que incluye múltiples modelos entrópicos. Cada uno de los múltiples modelos entrópicos del primer conjunto de modelos incluye un punto de conmutación de modelo para conmutar a un segundo conjunto de modelos, que incluye uno o varios modelos entrópicos. La herramienta procesa los símbolos utilizando el modelo entrópico seleccionado y entrega resultados del proceso.

10 Cada uno de los uno o varios modelos entrópicos del segundo conjunto de modelos puede incluir por sí mismo un punto de conmutación de modelo para conmutar a otro conjunto de modelos. Además, cada uno de los múltiples modelos entrópicos del primer conjunto de modelos puede incluir además un segundo punto de conmutación de modelo, para conmutar a otro conjunto de modelos. De manera más general, cada uno de los múltiples modelos entrópicos del primer conjunto de modelos puede incluir ninguno o más puntos de conmutación de modelos para conmutar a otro conjunto u otros conjuntos de modelos (incluyendo, por sí mismos, cada conjunto del otro conjunto o conjuntos de modelos, cero o más modelos entrópicos). De manera recursiva, para un conjunto de modelos dado del otro conjunto o conjuntos de modelos, el modelo o modelos entrópicos para dicho conjunto de modelos puede incluir cero o más puntos de conmutación de modelo, para conmutar a otro conjunto o conjuntos de modelos, y así sucesivamente.

20 De acuerdo con un segundo conjunto de técnicas y herramientas, un sistema genera modelos entrópicos. El sistema agrupa distribuciones de probabilidad de acuerdo con una primera métrica de costos (tal como el error cuadrático medio), lo que tiene como resultado grupos preliminares. El sistema refina los grupos preliminares de acuerdo con una segunda métrica de costos (tal como la entropía relativa), que es diferente respecto de la primera métrica de costos, lo que tiene como resultado grupos finales. A continuación, el sistema establece los modelos entrópicos en base, por lo menos en parte, a los grupos finales.

25 De acuerdo con un tercer conjunto de técnicas y herramientas, un sistema obtiene distribuciones de probabilidad para valores de símbolo. El sistema genera modelos entrópicos. Al hacerlo, el sistema limita múltiples valores de símbolo menos probables, a tener una distribución condicional común a través de las distribuciones de probabilidad, sin limitar de este modo múltiples valores de símbolo más probables.

30 Los anteriores y otros objetivos, características y ventajas de la invención resultarán más evidentes a partir de la siguiente descripción detallada, que hace referencia a las figuras adjuntas.

#### BREVE DESCRIPCIÓN DE LOS DIBUJOS

La figura 1 es un diagrama de bloques de un entorno operativo generalizado, junto con el cual pueden implementarse diversas realizaciones descritas.

35 Las figuras 2, 3, 4, 5, 6 y 7 son diagramas de bloques de codificadores y/o descodificadores generalizados, junto con los cuales pueden ser implementadas diversas realizaciones descritas.

Las figuras 8a y 8b son diagramas que muestran una señal de audio multicanal y una configuración de ventanas correspondiente, respectivamente.

Las figuras 9 y 10 son diagramas de bloques que muestran un codificador y un descodificador, respectivamente, con modulación de ruido temporal.

40 Las figuras 11 y 12 son diagramas de bloques que muestran un codificador y un descodificador, respectivamente, con predicción de coeficientes para la reducción de la velocidad de bits.

Las figuras 13 y 14 son diagramas de flujo que muestran técnicas para la predicción de coeficientes en codificación y descodificación, respectivamente, de coeficientes espectrales cuantificados.

45 Las figuras 15a y 15b son diagramas que muestran una señal de audio periódica en el dominio temporal y coeficientes espectrales correspondientes, respectivamente.

Las figuras 16 y 17 son diagramas de bloque que muestran un codificador y un descodificador, respectivamente, con reordenación de coeficientes.

Las figuras 18a hasta 18c son diagramas de flujo que muestran técnicas para reordenar coeficientes espectrales antes de la codificación entrópica.

Las figuras 19a hasta 19c son diagramas de flujo que muestran técnicas para reordenar coeficientes espectrales después de la decodificación entrópica.

5 La figura 20 es un diagrama que muestra los coeficientes espectrales de la figura 15b después de la reordenación.

La figura 21 es un diagrama que muestra una ganancia de codificación debida a la reordenación de coeficientes por subtrama, de un ejemplo de archivo de audio.

La figura 22 es un diagrama que muestra modelos entrópicos organizados jerárquicamente.

10 La figura 23 es un diagrama que muestra códigos de Huffman para distribuciones aproximadas de valores de símbolo.

Las figuras 24 y 25 son diagramas de flujo que muestran técnicas para agrupar vectores de aprendizaje para distribuciones de probabilidad.

La figura 26 es un diagrama de flujo que muestra una técnica para la codificación con el uso selectivo de múltiples modelos entrópicos.

15 La figura 27 es un diagrama de flujo que muestra una técnica para la decodificación con el uso selectivo de múltiples modelos entrópicos.

#### DESCRIPCIÓN DETALLADA

20 Se describen diversas técnicas y herramientas para codificación/decodificación y procesamiento asociado. Estas técnicas y herramientas facilitan la creación, distribución y reproducción de contenido de audio de alta calidad, incluso a velocidades de bits muy bajas.

Las diversas técnicas y herramientas descritas en el presente documento pueden ser utilizadas de manera independiente. Algunas de las técnicas y herramientas pueden ser utilizadas en combinación (por ejemplo, en diferentes fases de un proceso de codificación y/o decodificación combinado).

25 A continuación, se describen diversas técnicas haciendo referencia a diagramas de flujo de acciones de procesamiento. Las diversas acciones de procesamiento mostradas en los diagramas de flujo pueden ser consolidadas en menos acciones, o separadas en más acciones. Por simplicidad, a menudo no se muestra la relación de las acciones mostradas en un diagrama de flujo concreto, con las acciones descritas en otro lugar. En muchos casos, las acciones de un diagrama de flujo pueden ser reordenadas.

#### I. Ejemplos de entornos operativos para codificadores y/o decodificadores.

30 La figura 1 muestra un ejemplo generalizado de un entorno informático adecuado 100, en el que pueden implementarse varias de las realizaciones descritas. El entorno informático 100 no pretende insinuar ninguna limitación en relación con el alcance de la utilización o de la funcionalidad, puesto que las técnicas y herramientas descritas pueden ser implementadas en diversos entornos informáticos de propósito general o especializados.

35 Haciendo referencia a la figura 1, el entorno informático 100 incluye, por lo menos, una unidad 110 de proceso y una memoria 120. En la figura 1, esta configuración más básica 130 está contenida en una línea de trazos. La unidad 110 de proceso ejecuta instrucciones legibles por ordenador y puede ser un procesador real o virtual. En un sistema de multiprocesamiento, múltiples unidades de proceso ejecutan instrucciones legibles por ordenador para incrementar la potencia de procesamiento. La memoria 120 puede ser memoria volátil (por ejemplo registros, memoria caché, RAM), memoria no volátil (por ejemplo RAM, EEPROM, memoria flash, etc.) o alguna combinación  
40 de las dos. La memoria 120 almacena soporte lógico 180, que implementa un codificador y/o un decodificador que utiliza una o varias de las técnicas descritas en el presente documento.

45 Un entorno informático puede tener características adicionales. Por ejemplo, el entorno informático 100 incluye almacenamiento 140, uno o varios dispositivos de entrada 150, uno o varios dispositivos de salida 160, y una o varias conexiones de comunicación 170. Un mecanismo de interconexión (no mostrado) tal como un bus, un controlador o una red, interconecta los componentes del entorno informático 100. Habitualmente, un soporte lógico de sistema operativo (no mostrado) proporciona un entorno operativo para otro soporte lógico que se ejecuta en el entorno informático 100, y coordina actividades de los componentes del entorno de informático 100.

El almacenamiento 140 puede ser extraíble o no extraíble, e incluye discos magnéticos, cintas magnéticas o casetes, CD-ROMs, DVDs, o cualquier otro medio que pueda utilizarse para almacenar información y que sea accesible dentro del entorno informático 100. El almacenamiento 140 almacena instrucciones para el soporte lógico 180.

5 El dispositivo o dispositivos de entrada 150 pueden ser un dispositivo de entrada táctil tal como un teclado, un ratón, un lápiz o una rueda de desplazamiento, un dispositivo de entrada de voz, un dispositivo de escaneo, u otro dispositivo que proporcione entradas al entorno informático 100. Para codificación de audio o de vídeo, el dispositivo o dispositivos de entrada 150 pueden ser un micrófono, una tarjeta de sonido, una tarjeta de vídeo, una tarjeta de sintonizador de TV, o un dispositivo similar que admita entradas de audio o de vídeo en forma analógica o digital, o un CD-ROM o un CD-RW que lea muestras de audio o de vídeo en el entorno informático 100. El dispositivo o dispositivos de salida 160 pueden ser una pantalla, una impresora, un altavoz, un grabador de CDs, u otro dispositivo que proporcione una salida desde el entorno informático 100.

15 La conexión o conexiones de comunicación 170 permiten la comunicación, sobre un medio de comunicación, hasta otra entidad informática. El medio de comunicación transporta información tal como instrucciones ejecutables por ordenador, entradas o salidas de audio o de vídeo, u otros datos en una señal de datos modulada. Una señal de datos modulada es una señal que tiene una o varias de sus características establecidas o modificadas de tal modo que codifica la información en la señal. A modo de ejemplo, y no de limitación, un medio de comunicación incluye técnicas cableadas o inalámbricas implementadas con una portadora eléctrica, óptica, de RF, infrarroja, acústica u otra.

20 Las técnicas y herramientas pueden describirse en el contexto general de los medios legibles por ordenador. Los medios legibles por ordenador son cualquier medio disponible al que pueda accederse dentro de un entorno informático. A modo de ejemplo, y no de limitación, con el entorno informático 100, los medios legibles por ordenador incluyen la memoria 120, el almacenamiento 140, los medios de comunicación, y combinaciones de cualesquiera de los anteriores.

25 Las técnicas y herramientas pueden describirse en el contexto general de las instrucciones legibles ejecutables por ordenador, tales como las incluidas en módulos de programa, que son ejecutadas en un entorno informático en un procesador objetivo, real o virtual. En general, los módulos de programa incluyen rutinas, programas, bibliotecas, objetos, clases, componentes, estructuras de datos, etc., que llevan a cabo tareas concretas o implementan tipos concretos de datos abstractos. La funcionalidad de los módulos del programa puede combinarse o dividirse a conveniencia entre módulos de programa, en diversas realizaciones. Las instrucciones legibles por ordenador para módulos de programa pueden ser ejecutadas dentro de un entorno informático local o distribuido.

35 Para simplificar la presentación, la descripción detallada utiliza términos tales como "señalizar", "determinar" y "aplicar" para describir operaciones informáticas en un entorno informático. Estos términos son abstracciones de alto nivel para operaciones llevadas a cabo por un ordenador, y no deben confundirse con acciones realizadas por un ser humano. Las verdaderas operaciones informáticas correspondientes a estos términos pueden variar dependiendo de la implementación.

## II. Ejemplos de codificadores y descodificadores.

40 La figura 2 muestra un primer codificador 200 de audio en el cual pueden implementarse una o varias realizaciones descritas. El codificador 200 es un codificador 200 de audio perceptual, basado en transformadas. La figura 3 muestra un descodificador 300 de audio correspondiente.

La figura 4 muestra un segundo codificador 400 de audio, en el cual pueden implementarse una o varias realizaciones descritas. De nuevo, el codificador 400 es un codificador de audio perceptual, basado en transformadas, pero el codificador 400 incluye módulos adicionales para procesar audio multicanal. La figura 5 muestra un descodificador 500 de audio correspondiente.

45 La figura 6 muestra un codificador 600 de medios, más generalizado, en el que pueden implementarse una o varias realizaciones descritas. La figura 7 muestra un descodificador 700 de medios correspondiente.

50 Si bien los sistemas mostrados en las figuras 2 a 7 son generalizados, cada uno tiene características que se encuentran en sistemas del mundo real. En cualquier caso, las relaciones entre estos módulos dentro de los codificadores y descodificadores indican flujos de información en los codificadores y descodificadores; otras relaciones no se muestran por simplicidad. Dependiendo de la implementación y del tipo de compresión deseada, los módulos de un codificador o descodificador pueden ser añadidos, omitidos, divididos en múltiples módulos, combinados con otros módulos, y/o sustituidos con módulos similares. En realizaciones alternativas, codificadores y descodificadores con módulos diferentes y/o con otras combinaciones, procesan datos de audio o algún otro tipo de datos, de acuerdo con una o varias realizaciones descritas. Por ejemplo, en las figuras 2 a 7, los módulos que

procesan coeficientes espectrales pueden ser utilizados para procesar solamente coeficientes en uno o varios rangos secundarios de banda base o de frecuencia base (tal como a frecuencias inferiores), con diferentes módulos (no mostrados) procesando coeficientes espectrales en otros rangos secundarios de frecuencia (tales como frecuencias superiores).

5 A. Primer codificador de audio.

Globalmente, el codificador 200 recibe una serie temporal de muestras 205 de audio de entrada a ciertas velocidad y profundidad de muestreo. Las muestras 205 de audio de entrada son para audio multicanal (por ejemplo, estéreo) o audio mono. El codificador 200 comprime las muestras 205 de audio y multiplexa la información producida por los diversos módulos del codificador 200, para entregar un flujo de bits 295 en un formato tal como un formato WMA, un formato de transmisión continua avanzada ("ASF", Advanced Streaming Format ) u otro formato.

El transformador 210 de frecuencias recibe muestras 205 de audio y las convierte en datos en el dominio espectral. Por ejemplo, el transformador 210 de frecuencias divide las muestras 205 de audio en bloques, que pueden tener tamaño variable para permitir una resolución temporal variable. Los bloques pueden solaparse para reducir discontinuidades perceptibles entre bloques, que de lo contrario podría ser introducidas por una cuantificación posterior. El transformador 210 de frecuencias aplica a los bloques una transformada solapada modulada ("MLT", Modulated Lapped Transform), DCT modulada ("MDCT", modulated DCT), o alguna otra variedad de MLT ó DCT, o algún otro tipo de transformada de frecuencia modulada o no modulada, solapada o no solapada, o utiliza codificación de tren de ondas o sub-banda. El transformador 210 de frecuencia entrega bloques de datos de coeficientes espectrales, y entrega información complementaria tal como tamaños de bloque, al multiplexor ("MUX") 280.

Para datos de audio multicanal, el transformador 220 multicanal puede convertir los múltiples canales originales, codificados independientemente, en canales codificados conjuntamente. O bien, el transformador 220 multicanal puede pasar los canales izquierdo y derecho a través de canales codificados independientemente. El transformador 220 multicanal produce información complementaria para el MUX 280 indicando el modo de canal utilizado. El codificador 200 puede aplicar rematrización multicanal a un bloque de datos de audio después de una transformada multicanal.

El modelizador 230 de la percepción modela propiedades del sistema auditivo humano, para mejorar la calidad percibida de la señal de audio reconstruida, para una velocidad de bits dada. El modelizador 230 de la percepción utiliza cualquiera entre diversos modelos auditivos.

El modelizador 230 de la percepción entrega información que el dispositivo de ponderación 240 utiliza para modular el ruido de los datos de audio, con objeto de reducir la audibilidad del ruido. Por ejemplo, utilizando cualquiera entre diversas técnicas, el dispositivo 240 de ponderación genera factores de ponderación (en ocasiones, denominados factores de escala) para matrices de cuantificación (en ocasiones, denominadas máscaras) en base a la información recibida. A continuación, el dispositivo 240 de ponderación aplica los factores de ponderación a los datos recibidos desde el transformador multicanal 220. Un conjunto de factores de ponderación puede comprimirse para una representación más eficiente.

El cuantificador 250 cuantifica la salida del dispositivo 240 de ponderación, produciendo datos de coeficientes cuantificados para el codificador entrópico 260 e información complementaria que incluye el tamaño de la etapa de cuantificación para el MUX 280. En la figura 2, el cuantificador 250 es un cuantificador escalar uniforme, adaptativo. El cuantificador 250 aplica el mismo tamaño de etapa de cuantificación a cada coeficiente espectral, pero el propio tamaño de la etapa de cuantificación puede cambiar entre una iteración de un bucle de cuantificación y la siguiente, para afectar a la velocidad de bits de la salida del codificador entrópico 260. Otras clases de cuantificación son la cuantificación no uniforme, la cuantificación vectorial y/o la cuantificación no adaptativa.

El codificador entrópico 260 comprime sin pérdidas los datos de coeficientes cuantificados, recibidos desde el cuantificador 250, por ejemplo, llevando a cabo una codificación de serie-nivel y una codificación de longitud variable vectorial. En las secciones III hasta V se describen en detalle diversos mecanismos para codificación entrópica (potencialmente, incluyendo preprocesamiento) en algunas realizaciones. Alternativamente, el codificador entrópico 260 utiliza alguna otra forma de combinación de mecanismos de codificación entrópica. El codificador entrópico 260 puede calcular el número de bits gastados codificando información de audio, y pasar esta información al controlador 270 de velocidad/calidad.

El controlador 270 funciona junto con el cuantificador 250 para regular la velocidad de bits y/o la calidad de la salida del codificador 200. El controlador 270 entrega el tamaño de la etapa de cuantificación al cuantificador 250, con el objetivo de satisfacer restricciones de velocidad de bits y calidad.

Además, el codificador 200 puede aplicar sustitución de ruido y/o truncado de banda a un bloque de datos de audio.

El MUX 280 multiplexa la información complementaria recibida desde los otros módulos del codificador 200 de audio, junto con los datos sometidos a codificación entrópica, recibidos desde el codificador entrópico 260. El MUX 280 puede incluir una memoria tampón virtual que almacena el flujo de bits 295 a ser entregado por el codificador 200.

5 B. Primer descodificador de audio.

Globalmente, el descodificador 300 recibe un flujo de bits 305 de información de audio comprimida que incluye datos sometidos a codificación entrópica, así como información complementaria, a partir de la cual el descodificador 300 reconstruye muestras de audio 395.

10 El demultiplexor ("DEMUX") 310 realiza un análisis sintáctico de la información en el flujo de bits 305 y envía información a los módulos del descodificador 300. El DEMUX 310 incluye una o varias memorias intermedias para compensar las variaciones a corto plazo en la velocidad de bits debidas a fluctuaciones en la complejidad del audio, a la variabilidad temporal de la red, y/o a otros factores.

15 El descodificador entrópico 320 descomprime sin pérdidas códigos entrópicos recibidos desde el DEMUX 310, produciendo datos de coeficientes espectrales cuantificados. El descodificador entrópico 320 aplica habitualmente la inversa de la técnica de codificación entrópica utilizada en el codificador. En las secciones III a V se describen en detalle diversos mecanismos para la descodificación entrópica en algunas realizaciones.

20 El cuantificador inverso 330 recibe un tamaño de etapa de cuantificación procedente del DEMUX 310, y recibe datos de coeficientes espectrales cuantificados procedentes del descodificador entrópico 320. El cuantificador inverso 330 aplica el tamaño de etapa de cuantificación a los datos de coeficientes de frecuencia cuantificados, para reconstruir parcialmente los datos de coeficientes de frecuencia, o lleva a cabo de otro modo la cuantificación inversa.

Desde el DEMUX 310, el generador de ruido 340 recibe información que indica qué bandas en un bloque de datos están sustituidas con ruido así como cualesquiera parámetros para la forma del ruido. El generador 340 de ruido genera los modelos para las bandas indicadas, y pasa la información al dispositivo de ponderación inversa 350.

25 El dispositivo de ponderación inversa 350 recibe los factores de ponderación desde el DEMUX 310, modelos para cualesquiera bandas sustituidas con ruido desde el generador 340 de ruido, y los datos de coeficientes de frecuencia reconstruidos parcialmente, procedentes del cuantificador inverso 330. Cuando es necesario, el dispositivo de ponderación inversa 350 descomprime los factores de ponderación. El dispositivo de ponderación inversa 350 aplica los factores de ponderación a los datos de coeficientes de frecuencia reconstruidos parcialmente, para bandas que no han sido sustituidas con ruido. El dispositivo de ponderación inversa 330 añade a continuación los modelos de ruido recibidos desde el generador 340 de ruido para las bandas sustituidas con ruido.

35 El transformador multicanal inverso 360 recibe los datos de coeficientes espectrales reconstruidos, procedentes del dispositivo de ponderación inversa 350, y la información del modo de canal procedente del DEMUX 310. Si el audio multicanal está en canales codificados independientemente, el transformador multicanal inverso 360 pasa los canales. Si los datos multicanal están en canales codificados conjuntamente, el transformador multicanal inverso 360 convierte los datos en canales codificados independientemente.

El transformador inverso 370 de frecuencia recibe los datos de coeficientes espectrales entregados por el transformador multicanal 360, así como información complementaria tal como tamaños de bloque, procedente del DEMUX 380. El transformador inverso de frecuencia 370 aplica la inversa de la transformada de frecuencia utilizada en el codificador, y entrega bloques de muestras de audio reconstruidas 395.

40 C. Segundo codificador de audio.

45 Haciendo referencia a la figura 4, el codificador 400 recibe una serie temporal de muestras de audio de entrada 405, a cierta velocidad y profundidad de muestreo. Las muestras 405 de audio de entrada son para audio multicanal (por ejemplo, estéreo) o audio mono. El codificador 400 comprime las muestras de audio 405 y multiplexa la información producida por los diversos módulos del codificador 400 para entregar un flujo de bits 495 en un formato, tal como un formato WMA Pro u otro formato.

50 El codificador 400 selecciona entre múltiples modelos de codificación para las muestras de audio 405. En la figura 4, el codificador 400 conmuta entre un modo de codificación sin pérdidas mixto/puro y un modo de codificación con pérdidas. El modo de codificación sin pérdidas incluye el codificador sin pérdidas mixto/puro 472 y, habitualmente, se utiliza para compresión de alta calidad (y elevada velocidad de bits). El modo de codificación con pérdidas incluye componentes tales como el dispositivo de ponderación 442 y el cuantificador 460, y se utiliza habitualmente para compresión de calidad ajustable (y velocidad de bits controlada). La decisión de la selección depende de la entrada del usuario o de otros criterios.

- 5 Para la codificación con pérdidas, de datos de audio multicanal, el pre-procesador multicanal 410 realiza opcionalmente una rematrización de las muestras de audio 45 en el dominio temporal. En algunas realizaciones, el pre-procesador multicanal 410 realiza selectivamente una rematrización de las muestras de audio 405 para descartar uno o varios canales codificados, o incrementar la correlación entre canales en el codificador 400, mientras sigue permitiendo la reconstrucción (en alguna forma) en el descodificador 500. El pre-procesador multicanal 410 puede enviar al MUX 490 información complementaria, tal como instrucciones para el procesamiento posterior multicanal.
- 10 El módulo 420 de división en ventanas divide una trama de muestras 405 de entradas de audio en bloques de subtrama (ventanas). Las ventanas pueden tener tamaños variables en el tiempo y funciones de modulación de ventanas. Cuando el codificador 400 utiliza codificación con pérdidas, las ventanas de tamaño variable permiten una resolución temporal variable. El módulo 420 de división en ventanas entrega bloques de datos divididos, y entrega información complementaria, tal como tamaños de bloque, al MUX 490.
- 15 En la figura 4, el configurador 422 de mosaicos divide tramas de audio multicanal en un esquema por canal. El configurador 422 de mosaicos divide independientemente cada canal en la trama, si lo permite la calidad/velocidad de bits. Por ejemplo, el configurador 422 de mosaicos agrupa ventanas del mismo tamaño que son situadas conjuntamente en el tiempo, como un mosaico.
- 20 El transformador de frecuencia 430 recibe muestras de audio, y las convierte en datos en el dominio de frecuencias, aplicando una transformada tal como se ha descrito anteriormente para el transformador de frecuencia 210 de la figura 2. El transformador 430 de frecuencia entrega al dispositivo de ponderación 442 bloques de datos de coeficientes espectrales, y entrega al MUX 490 información complementaria, tal como tamaños de bloques. El transformador 430 de frecuencias entrega los coeficientes de frecuencia y la información complementaria al modelizador 440 de la percepción.
- 25 El modelizador 440 de la percepción modeliza propiedades del sistema auditivo humano, procesando datos de audio de acuerdo con un modelo auditivo.
- 30 El dispositivo de ponderación 442 genera factores de ponderación para matrices de cuantificación, en base a la información recibida desde el modelizador 440 de la percepción. El dispositivo de ponderación 442 aplica los factores de ponderación a los datos recibidos desde el transformador de frecuencia 430. El dispositivo de ponderación 442 entrega información complementaria, tal como matrices de cuantificación y factores de ponderación de canal, al MUX 490, y las matrices de cuantificación pueden ser comprimidas.
- 35 Para datos de audio multicanal, el transformador multicanal 450 puede aplicar una transformada multicanal. Por ejemplo, el transformador multicanal 450 aplica de manera selectiva y flexible la transformada multicanal, a parte de los canales y/o de las bandas de cuantificación en el mosaico, pero no a la totalidad. El transformador multicanal 450 utiliza selectivamente matrices predefinidas o matrices a medida, y aplica compresión eficiente a las matrices a medida. El transformador multicanal 450 produce información complementaria para el MUX 490 que indica, por ejemplo, las transformadas multicanal utilizadas y las partes de mosaico transformadas multicanal.
- 40 El cuantificador 460 cuantifica la salida del transformador multicanal 450, produciendo datos de coeficientes cuantificados para el codificador entrópico 470 e información complementaria que incluye tamaños de etapas de cuantificación, para el MUX 490. En la figura 4, el cuantificador 460 es un cuantificador escalar adaptativo, uniforme, que calcula un factor de cuantificación por mosaico, pero en lugar de esto el cuantificador 460 puede llevar a cabo alguna otra clase de cuantificación.
- 45 El codificador entrópico 460 comprime sin pérdidas datos de coeficientes cuantificados recibidos desde el cuantificador 460, en general tal como se ha descrito anteriormente haciendo referencia al codificador entrópico 260 de la figura 2. En las secciones III a V se describen en detalle diversos mecanismos para codificación entrópica (potencialmente, incluyendo preprocesamiento) en algunas realizaciones.
- 50 El controlador 480 funciona junto con el cuantificador 460 para regular la velocidad de bits y/o la calidad de la salida del codificador 400. El controlador 480 entrega los factores de cuantificación al cuantificador 460, con el objetivo de satisfacer restricciones de calidad y/o de velocidad de bits.
- El codificador sin pérdidas mixto/puro 472 y el codificador entrópico asociado 474 comprimen datos de audio para el modo de codificación sin pérdidas mixto/puro. El codificador 400 utiliza el modo de codificación sin pérdidas mixto/puro para una secuencia completa, o conmuta entre modos de codificación en un esquema trama a trama, bloque a bloque, o mosaico a mosaico.

El MUX 490 multiplexa la información complementaria recibida desde los otros módulos del codificador 400 de audio, junto con los datos sometidos a codificación entrópica, recibidos desde los codificadores entrópicos 470, 474. El MUX 490 comprende una o varias memorias intermedias para el control de velocidad, o para otros propósitos.

D. Segundo descodificador de audio.

- 5 Haciendo referencia a la figura 5, el segundo descodificador de audio 500 recibe un flujo de bits 505 de información de audio comprimida. El flujo de bits 505 incluye datos sometidos a codificación entrópica, así como información complementaria a partir de la cual el descodificador 500 reconstruye las muestras de audio 595.

10 El DEMUX 510 analiza sintácticamente la información en el flujo de bits 505 y envía información a los módulos del descodificador 500. El DEMUX 510 incluye una o varias memorias intermedias para compensar las variaciones a corto plazo en la velocidad de bits, debidas a fluctuaciones en la complejidad del audio, la variabilidad temporal de la red y/u otros factores.

15 El descodificador entrópico 520 descomprime sin pérdidas códigos entrópicos recibidos desde el DEMUX 510, habitualmente aplicando la inversa de las técnicas de codificación entrópica utilizadas en el codificador 400. Cuando se descodifican datos comprimidos en el modo de codificación con pérdidas, el descodificador entrópico 520 produce datos de coeficientes espectrales cuantificados. En las secciones III a V se describen en detalle diversos mecanismos para la descodificación entrópica en algunas realizaciones.

El descodificador sin pérdidas mixto/puro 522 y el descodificador o descodificadores entrópicos asociados 520 descomprimen sin pérdidas datos de audio codificados por el modo de codificación sin pérdidas mixto/puro.

20 El descodificador 530 de configuración de mosaicos recibe y, si es necesario, descodifica información que indica los modelos de mosaicos para tramas procedentes del DEMUX 590. La información de modelos de mosaico puede ser sometida a codificación entrópica o parametrizada de otro modo. A continuación, el descodificador 530 de configuración de mosaicos pasa la información del modelos de mosaico a algunos otros módulos del descodificador 500.

25 El transformador multicanal inverso 540 recibe del descodificador entrópico 520 los datos de coeficientes espectrales cuantificados, así como información del modelo de mosaico procedente del descodificador 530 de configuración de mosaicos, e información complementaria procedente del DEMUX 510 que indica, por ejemplo, la transformada multicanal utilizada y partes de mosaico transformadas. Utilizando esta información, el transformador multicanal inverso 540 descomprime, si es necesario, la matriz transformada, y aplica de manera selectiva y flexible una o varias transformadas multicanal inversas a los datos de audio.

30 El cuantificador inverso/dispositivo de ponderación inversa 550 recibe factores de cuantificación de mosaico y de canal así como matrices de cuantificación procedentes del DEMUX 510, y recibe datos de coeficientes espectrales cuantificados procedentes del transformador multicanal inverso 440. El cuantificador inverso/dispositivo de ponderación inversa 550 descomprime cuando es necesario la información de factor/matriz de cuantificación recibida, y a continuación lleva a cabo la ponderación y la cuantificación inversas.

35 El transformador inverso 560 de frecuencia recibe los datos de coeficientes espectrales entregados por el cuantificador inverso/dispositivo de ponderación inversa 550, así como información complementaria procedente del DEMUX 510, y la información del modelo de mosaico procedente del descodificador 530 de configuración de mosaicos. El transformador inverso 570 de frecuencia aplica la inversa de la transformada de frecuencia utilizada en el codificador, y entrega bloques al dispositivo de superposición/sumador 570.

40 Además de recibir información del modelo de mosaico procedente del descodificador 530 de configuración de mosaicos, el dispositivo de superposición/sumador 570 recibe información codificada procedente del transformador inverso 560 de frecuencia y/o del descodificador sin pérdidas mixto/puro 522. El dispositivo de superposición/sumador 570 superpone y suma datos de audio si es necesario, e intercala tramas u otras secuencias de datos de audio codificados con modos diferentes.

45 El post-procesador multicanal 580 realiza una rematrización de las muestras de audio en el dominio de temporal, entregadas por el dispositivo de superposición/sumador 570. Para un post-procesamiento controlado en flujo de bits, las matrices transformadas de post-procesamiento varían en el tiempo y son señalizadas o incluidas en el flujo de bits 505.

E. Codificador de medios generalizado.

La figura 6 muestra partes de un codificador de medios generalizado 600 que codifica audio, video, u otro contenido de medios. Por simplicidad, no se muestran numerosos módulos del codificador 600 y tipos de información complementaria que pueden depender del tipo de contenido de medios.

5 Tal como los codificadores 200, 400 mostrados en las figuras 2 y 4, respectivamente, el codificador 600 está basado en transformadas, asumiendo que la entrada mostrada en la figura 6 son coeficientes espectrales no cuantificados 605. Sin embargo, en algunas realizaciones uno o varios de los mecanismos de codificación entrópica descritos en el presente documento (por ejemplo, un mecanismo descrito en la sección V) se lleva a cabo para alguna otra clase de entrada.

10 El cuantificador 620 cuantifica los coeficientes 605, produciendo datos de coeficientes cuantificados. Por ejemplo, el cuantificador 620 es un cuantificador escalar adaptativo, uniforme, o alguna otra clase de cuantificador.

15 El preprocesador 640 de codificación entrópica lleva a cabo selectivamente un preprocesamiento previo a la codificación entrópica. Por ejemplo, el preprocesador 640 lleva a cabo predicción de coeficientes sobre coeficientes espectrales cuantificados, tal como se describe en la sección III. O bien, el preprocesador 640 reordena coeficientes espectrales cuantificados, tal como se describe en la sección IV. Alternativamente, el preprocesador 640 lleva a cabo algún otro tipo de preprocesamiento.

20 Además de coeficientes preprocesados, el preprocesador 640 entrega información complementaria que describe el preprocesamiento al flujo de bits de salida 695. Por ejemplo, la información complementaria incluye factores de predicción utilizados en la predicción de coeficientes, tal como se describe en la sección III. O bien, la información complementaria incluye información utilizada en la reordenación de coeficientes espectrales cuantificados, tal como se describe en la sección IV.

25 El codificador entrópico 660 comprime sin pérdidas datos de coeficientes cuantificados, por ejemplo, llevando a cabo codificación de serie-nivel y codificación vectorial de longitud variable. La sección V describe mecanismos para codificación entrópica adaptativa. Alternativamente, el codificador entrópico 660 utiliza alguna otra forma de combinación de mecanismos de codificación entrópica.

30 Si bien la figura 6 muestra simplemente el preprocesador 640 proporcionando una entrada al codificador entrópico 660, el codificador entrópico 660 proporciona alternativamente retroalimentación al preprocesador 640, la cual es utilizada por el preprocesador 640 para ajustar el preprocesamiento. Por ejemplo, el preprocesador 640 ajusta la reordenación de coeficientes en base a la retroalimentación procedente del codificador entrópico 660, de manera que la entrada al codificador entrópico 660 se ajusta mejor a un modelo de codificación entrópica.

F. Decodificador de medios generalizado.

La figura 7 muestra partes de un decodificador de medios generalizado 700 que descodifica audio, video, u otro contenido de medios. Por simplicidad, no se muestran numerosos módulos del decodificador 700 y tipos de información complementaria, que pueden depender del tipo de contenido de medios.

35 Tal como los decodificadores 300, 500 mostrados en las figuras 3 y 5 respectivamente, el decodificador 700 está basado en transformadas, asumiendo que la salida mostrada en la figura 7 son coeficientes espectrales reconstruidos 705. Sin embargo, en algunas realizaciones uno o varios de los mecanismos de descodificación entrópica descritos en el presente documento (por ejemplo, un mecanismo descrito en la sección V) se llevan a cabo para alguna otra clase de salida.

40 El decodificador entrópico 760 descomprime sin pérdidas datos de coeficientes cuantificados, por ejemplo, llevando a cabo descodificación de serie-nivel y descodificación vectorial de longitud variable. La sección V describe mecanismos para descodificación entrópica adaptativa. Alternativamente, el decodificador entrópico 760 utiliza alguna otra forma de combinación de mecanismos de descodificación entrópica.

45 El postprocesador 740 de descodificación entrópica lleva a cabo selectivamente un postprocesamiento después de la descodificación entrópica. Por ejemplo, el postprocesador 740 lleva a cabo predicción de coeficientes, sobre coeficientes espectrales cuantificados, tal como se describe en la sección III. O bien, el postprocesador 740 reordena coeficientes espectrales cuantificados, tal como se describe en la sección IV. Alternativamente, el postprocesador 740 lleva a cabo algún otro tipo de postprocesamiento.

50 Aparte de coeficientes sometidos a descodificación entrópica, el postprocesador 740 recibe información complementaria que describe el postprocesamiento, desde el flujo de bits 795. Por ejemplo, la información complementaria incluye factores de predicción utilizados en la predicción de coeficientes, tal como se describe en la

sección III. O bien, la información complementaria incluye información utilizada en la reordenación de coeficientes espectrales cuantificados, tal como se describe en la sección 4.

5 El cuantificador inverso 720 lleva a cabo cuantificación inversa, produciendo datos de coeficientes reconstruidos 705. Por ejemplo, el cuantificador inverso 720 es un cuantificador inverso escalar uniforme, adaptativo, o alguna otra clase de cuantificador.

III. Predicción de coeficientes en el dominio espectral para codificación y decodificación.

10 Un codificador de audio utiliza frecuentemente codificación de transformada, seguida por cuantificación y codificación entrópica para conseguir comprensión. Cuando se utiliza una transformada fija, para algunos modelos de señales de audio, permanece cierta correlación entre coeficientes adyacentes después de la transformada. A continuación se describen diversas técnicas y herramientas que explotan dicha correlación para mejorar la eficiencia de la codificación. En particular, en algunas realizaciones, un codificador tal como uno mostrado en las figuras 2, 4 ó 6, lleva a cabo predicción de coeficientes, sobre coeficientes espectrales cuantificados, durante la codificación. Un decodificador correspondiente (tal como uno mostrado en las figuras 3, 5 ó 7) lleva a cabo predicción de coeficientes, sobre coeficientes espectrales cuantificados, durante la decodificación.

15 **A. Ejemplo de dominio del problema.**

20 En un codificador de audio típico que comprime audio como una forma de onda, una señal de audio de entrada es transformada utilizando una MDCT de tamaño de entrada variable u otra transformada con una ventana de tamaño variable. Por ejemplo, supóngase que el análisis por división en ventanas del audio estéreo mostrado en la figura 8a tiene como resultado la configuración de ventanas mostrada en la figura 8b. En general, una configuración de ventanas de este tipo reduce el pre-eco y el post-eco en las señales codificadas (utilizando ventanas más cortas para segmentos transitorios), facilitando al mismo tiempo la eficiencia de codificación global (utilizando ventanas más largas para otros segmentos). Un objetivo del análisis por división en ventanas, es identificar límites de ventana, tales que la señal en el interior de cualquier ventana dada es fundamentalmente estacionaria.

25 Los coeficientes espectrales, antes o después de la transformada de canal, son cuantificados. Convencionalmente, se asume que los coeficientes espectrales de una subtrama u otra ventana no tienen ninguna correlación lineal entre ellos. En cambio, se asume que los coeficientes espectrales tienen habitualmente alguna relación estadística de orden superior, que los codificadores intentan explotar durante la codificación entrópica.

30 En la práctica, en diversas circunstancias no se cumplen varias hipótesis que están implícitas en dicha codificación. Por ejemplo, para ciertos tipos y modelos de señal de audio, los coeficientes espectrales para una subtrama u otra ventana no están necesariamente descorrelacionados. Por muchas de las mismas razones por las que una señal en una ventana puede ser no estacionaria (ver más abajo), los coeficientes espectrales pueden mostrar correlación lineal. Los actuales codificadores basados en la forma de onda no consiguen sacar una ventaja de dicha correlación en la codificación entrópica.

35 Como otro ejemplo, cuando se aplica análisis por división en ventanas a algunas señales de audio, la señal en el interior de la ventana concreta no es necesariamente estacionaria. Si la entrada de audio cambia sustancialmente con el tiempo (por ejemplo, para una señal de voz), incluso las ventanas cortas pueden ser insuficientes para aislar segmentos transitorios. O, si la memoria intermedia en un controlador de velocidad está llena, el controlador puede forzar al codificador a utilizar ventanas mayores para reducir la velocidad de bits, incluso si de lo contrario pudieran utilizarse ventanas menores. O, si la transición es lenta, el análisis por división en ventanas puede no conseguir detectar el transitorio, de tal modo que no se introducen ventanas más cortas. O bien, el análisis por división en ventanas puede proteger contra pre-eco introducido solamente por un transitorio por trama, y no por otros transitorios en la trama. O la señal en el interior de una ventana puede ser no estacionaria por alguna otra razón.

45 Los factores de escala pueden ayudar a controlar la distribución espectral de la distorsión. Sin embargo, en relación con la distribución temporal de la distorsión, la simple cuantificación sobre un espectro introduce una distorsión que es constante sobre un bloque completo transformado, lo que puede provocar una distorsión audible en segmentos temporales de la trama.

50 La modulación de ruido temporal ("TNS", Temporal Noise Shaping) es una tecnología de ciertas variantes de MPEG, que utiliza un enfoque predictivo en el dominio de frecuencias para modular el ruido de cuantificación en el tiempo. Con TNS, un codificador aplica un filtro de predicción a los coeficientes espectrales y cuantifica la señal filtrada, con objeto de limitar la ofuscación o el ruido de cuantificación a través de una ventana temporal completa. Las figuras 9 y 10 muestran TNS en un codificador y un decodificador, respectivamente.

Haciendo referencia a la figura 9, el codificador calcula la diferencia entre un coeficiente espectral no cuantificado 905 y un predictor, que es una combinación de dos coeficientes reconstruidos anteriores. Para la combinación, dos

5 coeficientes retardados temporalmente (en retardos 910 y 912), reconstruidos, son multiplicados cada uno por un factor de predicción 911, 913, y sumados entre sí. Los factores de predicción 911, 913 son cuantificados e incluidos en el flujo de bits 995. El cuantificador 970 cuantifica el valor diferencial, y el codificador entrópico 990 somete a codificación entrópica el valor diferencial cuantificado para entregarlo en el flujo de bits 995. El cuantificador inverso 980 reconstruye el valor diferencial y lo añade al predictor para el coeficiente 905. Esto tiene como resultado una reconstrucción del coeficiente, que es almacenado en memoria tampón en el primer retardo 910, y a continuación en el segundo retardo 912, para la contribución al predictor para un coeficiente de subsiguiente 905.

10 En el descodificador correspondiente, el descodificador entrópico 1090 somete a descodificación entrópica un valor diferencial procedente del flujo de bits 1095, y el cuantificador inverso 1080 realiza cuantificación inversa del valor diferencial. El descodificador combina el valor diferencial con un predictor, para producir un coeficiente espectral reconstruido 1005, en donde el predictor es una combinación de dos coeficientes reconstruidos previamente. El cálculo de la combinación involucra dos retardos 1010, 1012 y dos factores de predicción 1011, 1013, que son recuperados del flujo de bits 1095. El coeficiente espectral reconstruido 1005 es almacenado en memoria intermedia en el primer retardo 1010, y a continuación en el segundo retardo 1012, para su contribución al predictor para un coeficiente posterior 1005.

15 La TNS en la AAC MPEG permite hasta tres distintos filtros (o predictores) de respuesta infinita al impulso, a aplicar a diferentes regiones espectrales de una señal de entrada. Los coeficientes de los filtros son cuantificados e incluidos en el flujo de bits.

20 Incluso cuando la AAC de MPEG permite el uso de ventanas cortas, la TNS no se utiliza en las ventanas cortas, puesto que la información total requerida para la información de descripción del predictor es relativamente grande, lo que tiene como resultado menos bits para valores espectrales. De este modo, la TNS se permite solamente para ventanas largas en AAC de MPEG, lo que limita la utilidad de la TNS.

25 Asimismo, tal como se muestra en las figuras 9 y 10, la predicción en TNS se produce en el dominio no cuantificado/reconstruido. Como resultado, un descodificador tiene que intercalar operaciones de cuantificación inversa y predicción (y, posiblemente, incluso descodificación entrópica), lo que tiene como resultado una complejidad incrementada. Adicionalmente, para la predicción en el dominio no cuantificado/reconstruido, la operación TNS es especificada en AAC de MPEG como una operación de coma flotante, lo que provoca dificultades en implementaciones de punto fijo.

30 El predictor TNS es un predictor de segundo orden, que requiere dos multiplicaciones para la operación de predicción en cada coeficiente espectral. En el lado del codificador, el diseño de predictores eficaces puede ser difícil, y las predicciones inestables pueden constituir un problema.

35 Una arquitectura similar a la mostrada en las figuras 9 y 10, puede ser utilizada para la modulación diferencial por códigos de pulsos, en donde un codificador calcula la diferencia entre una muestra temporal y un predictor, y el predictor se basa en factores de predicción y muestras temporales cuantificadas inversas, almacenadas en memorias tampón. Habitualmente, la predicción utiliza un predictor detallado, que es difícil de diseñar y a menudo inestable, y que requiere señalización extensiva y lógica de reconstrucción. Además, la eficiencia de compresión de dichos escenarios no es buena.

40 En resumen, han sido descritos varios problemas que pueden ser tratados por técnicas y herramientas de predicción de coeficientes. Sin embargo, no es necesario aplicar dichas técnicas y herramientas de predicción de coeficientes, con objeto de tratar algunos o la totalidad de estos problemas.

**B. Ejemplos de arquitecturas para predicción de coeficientes.**

45 En algunas realizaciones, durante la codificación un codificador lleva a cabo la predicción de coeficientes sobre coeficientes espectrales cuantificados, y durante la descodificación un descodificador lleva a cabo la predicción de coeficientes sobre coeficientes espectrales cuantificados. En ciertos modelos y tipos de contenidos, la predicción de coeficientes reduce la redundancia en los coeficientes espectrales, con objeto de mejorar la eficiencia de la subsiguiente codificación entrópica. La predicción es reversible (durante la descodificación, la predicción de coeficientes (seguida a la descodificación entrópica) refleja la predicción de coeficientes en el codificador).

50 La figura 11 muestra un codificador con predicción de coeficientes espectrales cuantificados. Por ejemplo, el codificador es una versión modificada del codificador mostrado en la figura 2 ó 4, con etapas añadidas para calcular un predictor y un valor diferencial. O bien, el codificador es una versión modificada del codificador mostrado en la figura 6, con predicción de coeficientes como el preprocesamiento previo a la codificación entrópica.

Haciendo referencia a la figura 11, el codificador calcula la diferencia (denominada, asimismo, predicción residual) entre un coeficiente espectral cuantificado 1105 y un predictor. Para el predictor, un coeficiente espectral

cuantificado retardado temporalmente (en el retardo 1110) es multiplicado por un factor de predicción 1111. El factor de predicción 1111 es señalizado como información complementaria en el flujo de bits 1195. El codificador entrópico 1190 realiza codificación entrópica sobre el valor diferencial, para entregarlo en el flujo de bits 1195. El coeficiente espectral cuantificado 1105 es, asimismo, almacenado en memoria tampón en el primer retardo 1110, para el cálculo del predictor para un subsiguiente coeficiente espectral cuantificado 1105.

La figura 12 muestra un descodificador correspondiente, con predicción de los coeficientes espectrales cuantificados. Por ejemplo, el descodificador es una versión modificada del descodificador mostrado en la figura 3 ó 5, con etapas añadidas para calcular un predictor y combinar el predictor con un valor diferencial. O el descodificador es una versión modificada del descodificador mostrado en la figura 7, con predicción de coeficientes como postprocesamiento posterior a la descodificación entrópica.

Haciendo referencia a la figura 12, un descodificador entrópico 1290 descodifica un valor diferencial a partir del flujo de bits 1295. El descodificador calcula un predictor y combina el valor diferencial con el predictor, produciendo un coeficiente espectral cuantificado 1205. Para el predictor, un coeficiente espectral cuantificado retardado temporalmente (en el retardo 1210) es multiplicado por un factor de predicción 1211. El factor de predicción 1211 es sometido a análisis sintáctico desde el flujo de bits 1295. Asimismo, el coeficiente espectral cuantificado 1205 es almacenado en memoria intermedia en el primer retardo 1210, para el cálculo del predictor para un subsiguiente coeficiente espectral cuantificado 1205.

En las figuras 11 y 12, las operaciones de predicción y diferencial en el codificador y las operaciones de predicción y de suma en el descodificador, se producen en el dominio cuantificado. Esto simplifica el diseño del codificador y del descodificador y la complejidad, en la medida en que las operaciones se producen en el mismo dominio.

En algunas implementaciones, las operaciones de predicción, suma y resta, se producen sobre valores enteros. Habitualmente, esto simplifica la implementación puesto que las operaciones pueden ser realizadas con operaciones con enteros, frente a operaciones de coma flotante. Para simplificar más la predicción, puede cuantificarse un factor de predicción en un rango comprendido entre -1 y 1, utilizando un tamaño de etapa uniforme de 0,25. A continuación, las operaciones de multiplicación para el predictor pueden ser implementadas utilizando operaciones binarias de desplazamiento/suma.

En las figuras 11 y 12, el predictor es un predictor de primer orden, lo cual reduce de nuevo la complejidad del sistema del codificador/descodificador ("códec"). Con un predictor adaptativo de primer orden, el factor de predicción cambia, de manera que no es necesario utilizar el factor de predicción a largo plazo. Para un predictor de primer orden, un test de estabilidad es trivial. Por ejemplo, el codificador simplemente restringe el factor de predicción, a estar comprendido dentro del rango de -1 a +1, incluidos. Alternativamente, el predictor es un predictor de orden superior. Por ejemplo, el predictor tiene hasta 16 factores de predicción para un predictor de orden 16-ésimo.

Para la predicción adaptativa de coeficientes, el codificador cambia el factor de predicción entre subtrama y subtrama, o según otro esquema. Por ejemplo, el codificador divide una subtrama en múltiples segmentos de tamaño uniforme y calcula un factor de predicción por segmento. En relación con la señalización, el codificador señala el número de segmentos para la subtrama, así como los factores de predicción. Es decir, si una subtrama de 2048 coeficientes espectrales es dividida en 16 segmentos, el codificador señala el número de segmentos y un factor de predicción por cada segmento de 128 coeficientes. El número de segmentos por subtrama es señalizado una vez por secuencia, una vez por subtrama, o según otro esquema. Alternativamente, los elementos tienen longitudes variables y/o el codificador utiliza un mecanismo diferente para señalar factores de predicción (por ejemplo, señalizado solamente cambios en los factores de predicción, o señalización un factor de predicción y un número de segmentos para el cual es utilizado el factor de predicción).

Para algunas entradas, la predicción de coeficientes no mejora el rendimiento. Aparte de deshabilitar la predicción de coeficientes en un esquema segmento a segmento (descrito a continuación), un codificador y un descodificador pueden deshabilitar la predicción de coeficientes para una secuencia completa (por ejemplo, con un indicador de activación/desactivación de la capa de secuencia) o a algún otro nivel.

Cuando la predicción de coeficientes es utilizada para audio multicanal, la predicción de coeficientes se produce por canal codificado, cuando la cuantificación, etc., es posterior a la transformada multicanal durante la codificación. Durante la descodificación, la predicción de coeficientes se produce asimismo por canal codificado. De este modo, para dicho audio multicanal, la información de predicción que es señalizada por segmento o por subtrama es, habitualmente, señalizada por segmento o por subtrama de un canal codificado concreto. La predicción de canal puede deshabilitarse selectivamente por canal codificado, a nivel de secuencia o a algún otro nivel. Cuando se utiliza una predicción de coeficientes para audio multicanal, el número de segmentos por subtrama puede señalizarse por canal codificado, por subtrama de un canal codificado, o a algún otro nivel.

En algunos casos, la predicción de coeficientes proporciona una ganancia de codificación principalmente para coeficientes espectrales a frecuencias bajas y medias. Por lo tanto, la predicción de coeficientes puede deshabilitarse automáticamente para coeficientes espectrales a frecuencias superiores. O bien, si la ganancia de codificación a partir de la predicción de coeficientes es principalmente para coeficientes espectrales en rangos secundarios de frecuencia concretos, la predicción de coeficientes puede habilitarse selectivamente en dichos rangos secundarios de frecuencia y deshabilitarse en cualesquiera otros.

**C. Ejemplos de técnicas para predicción de coeficientes durante la codificación.**

La figura 13 muestra una técnica 1300 para la predicción de coeficientes espectrales cuantificados durante la codificación. Por ejemplo, un codificador tal como el mostrado en la figura 11 lleva a cabo la técnica 1300. Alternativamente, otro codificador lleva a cabo la técnica 1300.

Para empezar, el codificador calcula 1310 un factor de predicción para un segmento de audio. En general, el codificador calcula el factor de predicción utilizando cualquiera entre varias técnicas. Por ejemplo, para un predictor de primer orden, el codificador lleva a cabo una búsqueda exhaustiva de posibles factores de predicción con objeto de encontrar el factor de predicción final (por ejemplo, el factor que tiene como resultado un número menor de bits codificados entrópicamente). O bien, el codificador calcula una correlación constante para los coeficientes espectrales cuantificados del segmento (a saber,  $E\{x[i-1]x[i]\} / E\{x[i]x[i]\}$ ), para derivar el factor de predicción. O bien, para un predictor de orden superior, el codificador utiliza un algoritmo lineal de coeficientes de predicción (por ejemplo, que implique el cálculo de autocorrelación y autocovarianza) y no se requiere estabilidad. O bien, si el orden y la precisión del filtro son flexibles, el codificador calcula el orden del predictor (primero, segundo, tercero, etc.), y valores del factor de predicción y precisión para el segmento. Alternativamente, el codificador utiliza algún otro mecanismo para calcular el factor de predicción.

En muchos casos, los coeficientes espectrales cuantificados no presentan una correlación uniforme a través de todo el espectro de una subtrama. Para mejorar la precisión en dichas situaciones, el codificador puede cambiar el factor de predicción en un esquema espectral segmento a segmento. Por ejemplo, el codificador divide el espectro completo para una subtrama (u otro bloque de coeficientes espectrales) en múltiples segmentos de tamaño uniforme, y calcula un factor de predicción por segmento. Alternativamente, el codificador calcula un factor de predicción para un segmento que es el espectro completo de una subtrama u otro bloque de coeficientes espectrales, o divide el espectro de algún otro modo.

El codificador señala 1320 la información del factor de predicción para el segmento. Por ejemplo, el codificador cuantifica el factor de predicción y lo señala en un flujo de bits. El factor de predicción puede ser sometido a codificación entrópica. El codificador puede señalar un bit de activación/desactivación, como parte de la información del factor de predicción, con objeto de deshabilitar selectivamente la predicción de coeficientes en la decodificación, en una esquema segmento a segmento. La tabla 2 muestra representaciones de bit para un factor de predicción en una implementación en la cual los factores de predicción comprendidos en un rango de -1 a 1 están cuantificados utilizando un tamaño de etapa uniforme de 0,25.

Tabla 2. Representación del factor de predicción (información complementaria).

Factor de predicción	Representación binaria
-1,00	1000
-0,75	1001
-0,50	1010
-0,25	1011
0,00	0
0,25	1100
0,50	1101
0,75	1110
1,00	1111

Alternativamente, la información del factor de predicción es señalizada utilizando alguna otra representación.

Tal como se ha indicado anteriormente, es posible que no todos los segmentos se beneficien de la predicción de coeficientes espectrales. El factor de predicción de 0 deshabilita eficazmente la predicción para un segmento; la predicción no recibe ponderación y no tiene que ser calculada. Con los códigos mostrados en la tabla 2, el único símbolo de bit utilizado para señalar el factor de predicción de 0 actúa como un bit activación/desactivación para el

segmento afectado. Señalar el predictor cero con un solo bit, ahorra bits cuando el predictor cero es el factor de predicción más común.

5 Tal como se ha indicado anteriormente, se permiten predictores de orden superior. Para señalar información del factor de predicción para un predictor de orden superior, por ejemplo, el codificador envía en primer lugar la precisión y el orden del predictor, y a continuación envía los factores de predicción de uno en uno.

10 A continuación, el codificador determina 1330 si se utiliza, o no, predicción de coeficientes espectrales para el segmento. En caso afirmativo, el codificador predice 1340 dichos uno o varios coeficientes espectrales cuantificados en el segmento, y a continuación somete a codificación entrópica 1350 el coeficiente o coeficientes codificados predictivamente. Por ejemplo, el codificador utiliza memorias intermedias de retardo y aritmética, tal como se muestra en la figura 11, para la predicción de coeficientes. Alternativamente, el codificador utiliza algún otro mecanismo de predicción. (La predicción 1340 y la subsiguiente codificación entrópica 1350 pueden llevarse a cabo iterativamente para algunos tipos de codificación entrópica 1350, pero más habitualmente se realizan por lotes para codificación vectorial de longitud variable, codificación de serie-nivel, o algún otro tipo de codificación entrópica.)

15 Si el codificador se salta la predicción de coeficientes 1340, el codificador simplemente somete a codificación entrópica 1350 dichos uno o varios coeficientes espectrales cuantificados. Alternativamente, el codificador sigue el trayecto de codificación predictiva cuando el factor de predicción es 0.

A continuación, el codificador determina 1360 si continuar con el segmento siguiente o finalizar la técnica 1300. Si el codificador continúa, el codificador calcula 1310 el factor de predicción para el segmento siguiente, señala 1320 la información del factor de predicción, y así sucesivamente.

20 La figura 13 muestra el cálculo y la señalización de un factor de predicción en un esquema segmento a segmento, donde el número de segmentos está predeterminado y no es señalizado. Alternativamente, el número de segmentos para los cuales son calculados y señalizados los factores de predicción, es flexible. Habitualmente, esto mejora la precisión de la predicción, al costo de una sobrecarga de bits incrementada en la especificación de la información del segmento. Para una subtrama u otro bloque, el codificador encuentra una segmentación uniforme o no uniforme (por ejemplo, que tiene como resultado el número mínimo de bits), y el número total de segmentos y/u otra información de segmentación es señalizada en el flujo de bits.

#### D. Ejemplos de técnicas para la predicción de coeficientes durante la descodificación.

30 La figura 14 muestra una técnica 1400 para la predicción, durante la descodificación, de coeficientes espectrales cuantificados. Por ejemplo, un descodificador tal como el mostrado en la figura 12, lleva a cabo la técnica 1400. Alternativamente, otro descodificador lleva a cabo la técnica 1400.

35 Inicialmente, el descodificador obtiene 1410 información del factor de predicción para un segmento de audio. Por ejemplo, el descodificador analiza sintácticamente la información del factor de predicción, a partir de un flujo de bits, y reconstruye un factor de predicción. Si el factor de predicción está codificado entrópicamente, el descodificador somete a descodificación entrópica al factor de predicción. Si el codificador señala un bit de activación/desactivación como parte de la información del factor de predicción, con objeto de habilitar/deshabilitar selectivamente la predicción de coeficientes durante la descodificación, el descodificador obtiene el bit de activación/desactivación. De este modo, el descodificador puede cambiar el factor de predicción en un esquema segmento a segmento espectral, en donde el segmento es la totalidad o parte del espectro completo de una subtrama u otro bloque en función de la implementación, y donde la información del factor de predicción es señalizada utilizando cualquiera de los mecanismos descritos anteriormente haciendo referencia a la figura 13.

40 El descodificador entrópico descodifica 1420 información para uno o varios coeficientes espectrales cuantificados del segmento. Cuando durante la codificación ha sido utilizada predicción de coeficientes, la información consiste en uno o varios residuos de predicción (uno o varios valores diferenciales) para el o los coeficientes espectrales cuantificados. Cuando no se ha utilizado predicción de coeficientes durante la codificación (predictor cero), la información son los propios coeficientes espectrales cuantificados.

50 A continuación, el descodificador determina 1430 si se ha utilizado o no predicción de coeficientes espectrales para el segmento. En caso afirmativo, el descodificador predice 1440 el o los coeficientes espectrales cuantificados en el segmento. Por ejemplo, el descodificador utiliza memorias intermedias de retardo y aritmética, tal como se muestra en la figura 12, para la predicción de coeficientes. Alternativamente, el descodificador utiliza algún otro mecanismo de predicción. (La descodificación entrópica 1420 y la predicción 1440 pueden realizarse iterativamente para algunos tipos de descodificación entrópica 1420, pero más habitualmente son realizadas por lotes para descodificación vectorial de longitud variable, descodificación de serie-nivel, o algún otro tipo de descodificación entrópica.)

En algunos casos, el descodificador se salta la predicción de coeficientes durante la descodificación, simplemente sometiendo a descodificación entrópica 1420 el o los coeficientes espectrales cuantificados. Alternativamente, el descodificador sigue el trayecto de descodificación predictiva cuando el factor de predicción es 0.

5 A continuación, el descodificador determina 1450 si continuar con el segmento siguiente o finalizar la técnica 1400. Si el descodificador continua, el descodificador obtiene 1410 la información del factor de predicción para el segmento siguiente, y así sucesivamente.

En la figura 14, el número de segmentos está predeterminado y no es señalizado. Alternativamente, el número de segmentos y factores de predicción es flexible, y el descodificador analiza sintácticamente la información de segmentación señalizada por el codificador.

10 E. Resultados.

En general, la predicción de coeficientes espectrales cuantificados mejora la eficiencia de la codificación entrópica subsiguiente, para ciertos tipos y patrones de contenidos. Por ejemplo, la predicción reduce la redundancia entre coeficientes adyacentes, haciendo más eficiente la subsiguiente codificación vectorial de longitud variable y/o codificación de serie-nivel. Por contraste, el propósito de MPEG TNS es controlar la distribución temporal de la distorsión.

15 Para medir la mejora en la eficiencia de la codificación, debida a la predicción de los coeficientes espectrales cuantificados, se codificó un gran conjunto de canciones de prueba utilizando predicción de coeficientes. Para una típica canción de entrada, la mayor parte de las subtramas en la canción no obtuvieron ningún beneficio utilizando predicción de coeficientes en el dominio cuantificado, sin bien algunas subtramas se beneficiaron de manera muy sustancial. Por ejemplo, los bits producidos para algunas subtramas descendieron tanto como al 30% con predicción de coeficientes espectrales cuantificados. Para algunas canciones, la reducción global de la velocidad de bits con predicción de coeficientes fue del 3% trabajando a una velocidad de bits nominal de 32 Kb/s, y la reducción global de la velocidad de bits fue del 3,75% a 128 Kb/s. Para el conjunto completo de canciones, la reducción global de la velocidad de bits estuvo en torno al 0,5%.

25 Si bien muchos tipos de predicción utilizan un predictor de orden superior o una precisión superior para conseguir ganancia de codificación, un predictor de primer orden con precisión relativamente baja (por ejemplo, 3 bits por valor de factor de predicción cuantificado) funciona muy bien en coeficientes espectrales cuantificados en la mayor parte de los escenarios. Habitualmente, los coeficientes espectrales cuantificados son enteros muy pequeños, de manera que incrementar la precisión del factor de predicción no cambia necesariamente el valor predicho o no lo mejora (el valor residual es un entero para codificación entrópica, y calcular el valor predicho como un entero es aceptable).

30 Además, incluso cuando existe una correlación de orden superior en los coeficientes espectrales, habitualmente la correlación de orden superior es distorsionada por la cuantificación, de manera que no son necesarios predictores de orden superior.

35 Sin embargo, en algunos escenarios, cuando los tamaños de la etapa de cuantificación son pequeños y los coeficientes espectrales cuantificados tienen grandes amplitudes, predictores de orden superior y/o factores de predicción de precisión superior pueden tener como resultado mayores mejoras en la eficiencia de la codificación. Las técnicas y herramientas de predicción de coeficientes descritas anteriormente soportan predictores de orden superior y factores de predicción de precisión superior, en una forma general.

IV. Intercalación o reordenación de coeficientes espectrales.

40 Tal como se indicado previamente, un codificador de audio utiliza a menudo codificación de transformada seguida por cuantificación y codificación entrópica, para conseguir comprensión. Para algunos patrones de señales de audio, permanece un patrón periódico en los coeficientes espectrales después de la transformada de frecuencia. Se describen diversas técnicas y herramientas para explotar dicha redundancia con objeto de mejorar la eficiencia de codificación. En particular, en algunas realizaciones, un codificador tal como el mostrado en las figuras 2, 4 ó 6, lleva a cabo intercalación o reordenación de los coeficientes espectrales cuantificados. Un descodificador correspondiente (tal como uno mostrado en las figuras 3, 5 ó 7) invierte la intercalación o reordenación de los coeficientes espectrales cuantificados.

**A. Ejemplo de dominio del problema.**

50 Convencionalmente, se asume que los coeficientes espectrales de una subtrama u otra ventana no tienen ninguna correlación lineal entre ellos. En cambio, se asume que los coeficientes espectrales tienen habitualmente alguna relación estadística de orden superior, la cual intentan explotar los codificadores durante la codificación entrópica.

Estas hipótesis no se sostienen en ciertas circunstancias. Para ciertos tipos de patrones de señales de audio, los coeficientes espectrales para una subtrama u otra ventana no están necesariamente no correlacionados. Esto ocurre, por ejemplo, cuando una señal de audio es periódica en el dominio temporal, y los coeficientes espectrales de la señal periódica muestran, asimismo, periodicidad. En la práctica, las señales sinusoidales muestran a menudo este comportamiento, así como ciertas señales no estacionarias.

A modo ilustrativo, la figura 15a muestra una señal de audio periódica en el dominio temporal, que registra amplitudes para una serie temporal de muestras. La figura 15b muestra los coeficientes espectrales cuantificados correspondientes, a partir de una operación de DCT. En la figura 15b, hay potentes coeficientes espectrales máximos, distintos de cero, en torno a cada 57 coeficientes espectrales, y los coeficientes espectrales en otras posiciones tienen fundamentalmente un valor cero o menor. Codificar entrópicamente directamente coeficientes espectrales con esta clase de patrón periódico utilizando técnicas tales como codificación de serie-nivel o codificación vectorial de longitud variable, no es eficiente. En particular, codificar un coeficiente máximo con coeficientes de valor cero o de valores pequeños en torno al mismo, utiliza habitualmente muchos bits tanto en codificación de serie-nivel como en codificación vectorial de longitud variable. Sin embargo, este tipo de patrón de máximos es común para señales periódicas.

En resumen, han sido descritos varios problemas que pueden ser tratados mediante técnicas y herramientas de reordenación de coeficientes. Sin embargo, no es necesario aplicar dichas técnicas y herramientas de reordenación de coeficientes para tratar algunos o la totalidad de estos problemas.

**B. Ejemplos de arquitecturas para reordenar coeficientes espectrales.**

En algunas realizaciones, un codificador lleva a cabo reordenación sobre los coeficientes espectrales cuantificados antes de la codificación entrópica, y un decodificador correspondiente lleva a cabo reordenación sobre los coeficientes espectrales cuantificados después de la decodificación entrópica. Para ciertos patrones y tipos de contenidos tales como señales periódicas con tonos o armónicos, la reordenación reduce la redundancia en los coeficientes espectrales para mejorar la eficiencia de la subsiguiente codificación entrópica. Durante la decodificación, la reordenación (siguiente a la decodificación entrópica) compensa la reordenación en el codificador.

La figura 16 muestra un codificador con reordenación de los coeficientes espectrales cuantificados. Por ejemplo, el codificador es una versión modificada del codificador mostrado en las figuras 2 ó 4, con etapas añadidas para reordenar coeficientes espectrales. O bien, el codificador es una versión modificada del codificador mostrado en la figura 6, con reordenación como el preprocesamiento previo a la codificación entrópica.

Haciendo referencia a la figura 16, el codificador recibe los coeficientes espectrales cuantificados 1605 desde un cuantificador. Los coeficientes espectrales cuantificados son procesados por el módulo de reordenación/intercalación 1680, que reordena opcionalmente parte o la totalidad de los coeficientes espectrales 1605, señalizando la información de reordenación en el flujo de bits 1695.

Supóngase que los coeficientes espectrales cuantificados 1605 presentan un patrón periódico que puede ser explotado para mejorar la eficiencia de la codificación entrópica. Antes de la codificación entrópica, los coeficientes espectrales cuantificados son intercalados o reordenados considerando la periodicidad de los coeficientes. Por ejemplo, la reordenación agrupa conjuntamente coeficientes máximos de valor superior, lo que mejora la eficiencia de la subsiguiente codificación vectorial de longitud variable para dichos coeficientes, y la reordenación agrupa conjuntamente otros coeficientes (por ejemplo, coeficientes de valor cero y coeficientes de valor reducido entre máximos), lo que mejora la eficiencia de la subsiguiente codificación de serie-nivel para dichos coeficientes.

Para intercalar coeficientes espectrales, el codificador intercala los coeficientes espectrales a lo largo del segmento que muestra el patrón periódico. Como un simple ejemplo, el codificador explora a través de los coeficientes en los periodos, en un modo de múltiples pasadas, seleccionando en primer lugar los primeros coeficientes en los periodos respectivos, seleccionando a continuación los segundos coeficientes en los periodos respectivos, seleccionando después los terceros coeficientes en los periodos respectivos, y así sucesivamente. El codificador prosigue la reordenación hasta que todos los coeficientes han sido seleccionados. Supóngase que una serie de coeficientes espectrales incluyen cuatro periodos A, B, C y D, y que cada periodo contiene cuatro coeficientes espectrales. Antes de la intercalación, la serie es:

A<sub>0</sub> A<sub>1</sub> A<sub>2</sub> A<sub>3</sub> B<sub>0</sub> B<sub>1</sub> B<sub>2</sub> B<sub>3</sub> C<sub>0</sub> C<sub>1</sub> C<sub>2</sub> C<sub>3</sub> D<sub>0</sub> D<sub>1</sub> D<sub>2</sub> D<sub>3</sub>,

y después de la intercalación, la serie es:

A<sub>0</sub> B<sub>0</sub> C<sub>0</sub> D<sub>0</sub> A<sub>1</sub> B<sub>1</sub> C<sub>1</sub> D<sub>1</sub> A<sub>2</sub> B<sub>2</sub> C<sub>2</sub> D<sub>2</sub> A<sub>3</sub> B<sub>3</sub> C<sub>3</sub> D<sub>3</sub>.

De este modo, la serie reordenada sitúa los coeficientes 0, 4, 8 y 12 en primer lugar, a continuación los coeficientes 1, 5, 9 y 13, y así sucesivamente. Si, en cada periodo, solamente el primer coeficiente tiene un valor significativo, después de la intercalación solamente los primeros cuatro coeficientes en la serie tienen valores significativos, y todos los demás coeficientes tienen un valor pequeño o un valor cero. La codificación vectorial de longitud variable comprime de manera eficiente los primeros cuatro coeficientes, y la codificación de serie-nivel trata eficientemente el resto.

Volviendo a la figura 16, después de la reordenación opcional 1680, el codificador entrópico 1690 codifica entrópicamente los coeficientes espectrales (potencialmente reordenados). El codificador señala la información sometida a codificación entrópica, en el flujo de bits 1695.

La figura 17 muestra un descodificador correspondiente con reordenación de los coeficientes espectrales cuantificados. Por ejemplo, el descodificador es una versión modificada del descodificador mostrado en las figuras 3 ó 5, con etapas añadidas para la reordenación. O bien, el descodificador es una versión modificada del descodificador mostrado en la figura 7, con reordenación como el postprocesamiento posterior a la descodificación entrópica.

Haciendo referencia a la figura 17, el descodificador entrópico 1790 descodifica información para el coeficiente espectral cuantificado desde el flujo de bits 1795. Utilizando información de reordenación analizada sintácticamente desde el flujo de bits 1795, el módulo de reordenación/intercalación 1780 reordena opcionalmente parte o la totalidad de los coeficientes espectrales descodificados, produciendo los coeficientes espectrales cuantificados 1705 en el orden original. Esencialmente, la reordenación en el descodificador invierte la reordenación realizada en el codificador.

En la serie de ejemplo mostrada anteriormente, se lleva a cabo una simple reordenación basada en una longitud de periodo. Sin embargo, en algunos casos una simple reordenación de este tipo no tiene en cuenta información no periódica importante en un segmento, ceros importantes u otros desplazamientos en periodos específicos, y/o el agrupamiento de coeficientes máximos en los comienzos de los periodos. Información adicional de reordenación (descrita a continuación) puede tratar estos fenómenos. Para proporcionar un ejemplo numérico simple, supóngase que un segmento tiene 128 coeficientes espectrales e incluye un patrón periódico para algunos de los coeficientes. El patrón del período tiene una longitud de período promedio de 10 coeficientes, comienza en el coeficiente 19-ésimo, y finaliza en el coeficiente 102-ésimo. En términos de múltiplos de la longitud del período, como una estimación aproximada, el primer periodo reordenado es el tercer periodo (coeficientes 20-29) del segmento, y el último período reordenado es el décimo periodo (coeficientes 90-99). El desplazamiento para el tercer periodo es -1 (que indica una posición de comienzo para el período en el coeficiente 19-ésimo en lugar del 20-ésimo), y el desplazamiento para el décimo periodo es 2. Pueden señalizarse asimismo desplazamientos para otros periodos, cuando proceda. Si los periodos a reordenar comienzan típicamente con múltiples coeficientes máximos, puede señalizarse un valor para indicar el número de coeficientes iniciales por periodo que deberían mantenerse adyacentes incluso después de la reordenación.

Para la reordenación de coeficientes adaptativos, el codificador cambia la reordenación de subtrama a subtrama, o sobre alguna otra base. Por ejemplo, el codificador divide una subtrama en múltiples segmentos y calcula la información de reordenación para uno o varios de los segmentos, señalizando la información de segmentación así como la información de reordenación. Alternativamente, el codificador utiliza un mecanismo diferente para segmentación y/o señalización.

Para algunas entradas, la reordenación de coeficientes no mejora el rendimiento. Aparte de deshabilitar la reordenación de coeficientes en un esquema segmento a segmento (descrito a continuación), un codificador y un descodificador pueden deshabilitar la reordenación de coeficientes para una secuencia completa (por ejemplo, con un indicador de activación/desactivación de la capa de secuencia) o a algún otro nivel.

Cuando la reordenación de coeficientes es utilizada para audio multicanal, la reordenación de coeficientes se produce por canal codificado, cuando la cuantificación, etc., ocurre más abajo que la transformada multicanal durante la codificación. Durante la descodificación, la reordenación de coeficientes se produce asimismo por canal codificado. De este modo, para audio multicanal de este tipo, la información de reordenación que es señalizada por segmento, por subtrama, o por periodo, es habitualmente señalizada por segmento, por subtrama o por periodo, para un canal codificado concreto. Cuando la reordenación de coeficientes se utiliza para audio multicanal, la información de segmentación y la información de activación/desactivación de reordenación puede ser señalizada por canal codificado, por subtrama de un canal codificado, o a algún otro nivel.

En muchos casos, la reordenación de coeficientes proporciona ganancias de codificación principalmente para coeficientes espectrales en frecuencias bajas y medias. Por lo tanto, la reordenación de coeficientes puede deshabilitarse automáticamente para coeficientes espectrales a frecuencias superiores. O bien, si la ganancia de codificación a partir de la reordenación de coeficientes es principalmente para coeficientes espectrales en rangos

secundarios de frecuencia concretos, la reordenación de coeficientes puede habilitarse selectivamente en dichos rangos secundarios de frecuencia y deshabilitarse en cualesquiera otros.

5 La predicción de coeficientes descrita en la sección III puede ser utilizada junto con la reordenación de coeficientes, pero la predicción de coeficientes y la reordenación de coeficientes se utilizan de manera más habitual por separado, para diferentes categorías de entradas. Cuando se utilizan conjuntamente, la predicción de coeficientes sigue a la reordenación durante la codificación, y la reordenación de coeficientes sigue a la predicción durante la descodificación, y la predicción de coeficientes se utiliza, por lo menos, sobre parte (por ejemplo, coeficientes de pico) de los coeficientes reordenados.

### C. Ejemplos de técnicas para reordenar coeficientes durante la codificación.

10 La figura 18a muestra una técnica 1800 para reordenar los coeficientes espectrales cuantificados durante la codificación, y las figuras 18b y 18c detallan posibles modos de llevar a cabo ciertas acciones de la técnica 1800. Por ejemplo, un codificador tal como el mostrado en la figura 16 lleva a cabo la técnica 1800. Alternativamente, otro codificador lleva a cabo la técnica 1800.

15 Inicialmente, el codificador 1810 calcula información de reordenación para un segmento. Por ejemplo, el codificador calcula 1810 la información de reordenación, tal como se muestra en la figura 18b. Alternativamente, el codificador calcula información de reordenación diferente y/o adicional.

20 Haciendo referencia a la figura 18b, el codificador identifica 1812 un segmento en cuyo interior se reordenarán los coeficientes. Por ejemplo, el codificador encuentra un segmento de los coeficientes espectrales que tiene un patrón periódico. Con fines ilustrativos, en la figura 15b, solamente los primeros 800 coeficientes más o menos tienen un patrón periódico.

25 El codificador puede excluir de la reordenación algunos periodos del segmento. Por ejemplo, si los primeros uno o dos periodos no se parecen a los otros periodos, los primeros uno o dos periodos son excluidos del proceso de reordenación. En algunos casos, la primera parte del segmento incluye coeficientes no periódicos o ceros importantes. Así, el codificador hace un seguimiento del primer periodo a reordenar en el segmento. Igualmente, el codificador hace asimismo un seguimiento del último periodo a reordenar en el segmento.

30 A continuación, el codificador identifica 1814 la longitud del periodo del segmento. Por ejemplo, el codificador cuenta el número de máximos en el segmento y divide la longitud del segmento por el número de máximos. O bien, el codificador lleva a cabo una búsqueda exhaustiva de longitudes de periodo candidatas. O bien, el codificador busca longitudes de periodo candidatas utilizando un enfoque de refinamiento binario (a diferencia de una búsqueda exhaustiva del espacio de parámetros). O el codificador evalúa longitudes de series de coeficientes de valor cero/valor pequeño. O bien, el codificador utiliza algún otro mecanismo para identificar la longitud de periodo para el segmento. La longitud del periodo puede limitarse a valores enteros, o la longitud del periodo puede ser asimismo un valor no entero. Permitir una precisión por debajo de los enteros puede mejorar significativamente la eficiencia de la reordenación, mejorando finalmente la ganancia de la codificación entrópica.

35 Asimismo, el codificador identifica 1816 otra información de reordenación, que puede incluir ajustes de periodos y valores de preámbulo. Por ejemplo, en una implementación que permite longitudes de periodo no enteras, el codificador calcula otra información de reordenación, como sigue.

40 La posición de comienzo inicial del periodo  $i$  es  $\text{redondear}(i * \text{longitud\_del\_periodo})$ , y la posición de finalización inicial del periodo  $i$  es la posición de comienzo inicial del periodo siguiente. El codificador mantiene una lista de posiciones de periodos, que almacena las posiciones de comienzo y/o las posiciones finales de los periodos, con fines de seguimiento. Asimismo, esto permite al codificador ajustar simplemente las posiciones de los periodos en la lista cuando evalúa diferentes posiciones.

45 En particular, el codificador puede desplazar la posición de comienzo y/o la posición de finalización del periodo, en uno o varios coeficientes desde la posición inicial, con objeto de mejorar la codificación entrópica. Por ejemplo, si existen varios coeficientes grandes, significativos, inmediatamente antes de la posición de comienzo inicial del periodo, el codificador desplaza un par de coeficientes a la izquierda la posición de comienzo, de manera que dichos coeficientes grandes, significativos, aparecen al principio del periodo en lugar de al final del periodo anterior. Alternativamente, el codificador utiliza algún otro mecanismo para determinar cantidades de ajuste para las posiciones de comienzo y/o finalización de los periodos a reordenar.

50 Asimismo, el codificador elige un valor de preámbulo. Preámbulo, indica coeficientes al comienzo de un periodo que no son reordenados entre sí. Habitualmente, el pico al comienzo de un periodo no es un solo coeficiente espectral. Puede haber dos o tres coeficientes con valores grandes al comienzo del periodo, por ejemplo, y dichos coeficientes son coeficientes de preámbulo. Los coeficientes de preámbulo son intercalados de manera especial, siendo tratados

eficazmente como un grupo para la reordenación. En otras palabras, los coeficientes de preámbulo son adyacentes incluso después de la reordenación para los periodos de segmento. El valor de preámbulo indica el número de coeficientes de preámbulo (por ejemplo, 1, 2, 3) para los periodos a reordenar. O bien, en lugar de calcular el preámbulo por segmento, el codificador calcula el preámbulo por período a reordenar.

5 Alternativamente, el codificador utiliza algún otro mecanismo para identificar 1816 la otra información de reordenación.

Volviendo a la figura 18a, el codificador señala 1830 la información de reordenación para el segmento, en el flujo de bits. Por ejemplo, el codificador señala 1830 la información de reordenación tal como se muestra en la figura 18c, para reordenar la información calculada tal como se muestra en la figura 18b. Alternativamente, el codificador  
10 señala información de reordenación diferente y/o adicional.

Haciendo referencia a la figura 18c, el codificador señala 1832 un bit de activación/desactivación para la reordenación. Por ejemplo, el codificador compara el costo de bits cuando se utiliza reordenación de coeficientes, con el costo de bits cuando no se utiliza reordenación de coeficientes. El codificador selecciona el modo que proporciona un mejor rendimiento, y el codificador utiliza en un único bit por segmento para indicar qué modo se  
15 selecciona. Alternativamente, el codificador señala información de activación/desactivación utilizando algún otro mecanismo y/o durante alguna duración diferente a un segmento completo.

Cuando se utiliza reordenación (la salida "sí" de la decisión 1834), el codificador señala 1836 la longitud del periodo. Cuando se permiten longitudes de periodo no enteras, la longitud del periodo puede representarse con una parte entera y una parte fraccionaria, ambas señalizadas en el flujo de bits. Una longitud de periodo entera (o la parte entera de una longitud de periodo no entera) es señalizada como un código de longitud fija ["FLC", fixed length code] con  $\log_2(\text{longitud\_de\_período\_más\_larga})$  bits. Por ejemplo, la longitud de periodo más larga es 128 y la longitud del periodo entera es señalizada con  $\log_2(128) = 7$  bits. Una parte fraccionaria puede señalizarse con un FLC de tres bits. Alternativamente, la longitud del período es señalizada con otro mecanismo.

Asimismo, el codificador 1838 señala el primer periodo para el cual serán reordenados los coeficientes. De hecho, esto indica aproximadamente la posición del comienzo para la reordenación. El primer período reordenado puede representarse en unidades de la longitud del periodo. El primer periodo reordenado es señalizado, por ejemplo, con un FLC de tres bits, en cuyo caso el primer periodo reordenado es algún periodo entre el primer período y el octavo periodo en el segmento. Alternativamente, el primer periodo reordenado es señalizado con otro mecanismo.

Asimismo, el codificador 1840 señala el último periodo para el cual serán reordenados los coeficientes. El último período reordenado puede representarse en unidades de la longitud del periodo. El último periodo reordenado es señalizado, por ejemplo, como un FLC con  $\log_2(\text{máximo\_número\_de\_periodos})$  bits. El codificador obtiene el número máximo de periodos a partir del número de coeficientes en el segmento y la longitud del periodo. Alternativamente, el último periodo reordenado es señalizado con otro mecanismo.

El codificador señala 1842 ajustes de posición. Para los periodos para los cuales se reordenarán los coeficientes, el codificador señala información que indica desplazamientos en relación con las posiciones iniciales de comienzo y/o de finalización. Por ejemplo, se señala un valor de ajuste por periodo, y el valor de ajuste es señalizado como un número de coeficientes. Dicho valor de ajuste puede ser señalizado como un FLC con  $\log_2(\text{rango\_de\_desplazamiento})$  bits. Por lo tanto, si el valor de desplazamiento es 16, el valor de ajuste es señalizado con  $\log_2(16) = 4$  bits, con un rango de ajuste de -8 ... 7 coeficientes. Alternativamente, el valor de ajuste es señalizado con otro mecanismo (por ejemplo, señalizando ajustes en relación con valores de ajuste previos (no en términos absolutos), o señalizando un ajuste para todos los periodos).

Asimismo, el codificador señala 1844 un valor de preámbulo. Un valor de preámbulo de algún número de coeficientes es señalizado como un FLC con  $\log_2(\text{preámbulo\_mayor} + 1)$  bits. Por ejemplo, la longitud del preámbulo más largo es de 3 (para un preámbulo de 0, 1, 2 ó 3), y el valor de preámbulo es señalizado con  $\log_2(4) = 2$  bits.  
45 Alternativamente, los valores de preámbulo son señalizados con otro mecanismo.

Volviendo a la figura 18a, el codificador determina 1860 si se utiliza o no reordenación de coeficientes. En caso negativo, el codificador simplemente realiza codificación entrópica 1880 de los coeficientes espectrales cuantificados del segmento, utilizando codificación vectorial de longitud variable, codificación de serie-nivel, o alguna otra codificación entrópica. Por otra parte, si se utiliza reordenación de coeficientes, el codificador reordena 1870, por lo menos, parte de los coeficientes del segmento y realiza codificación entrópica 1880 sobre los coeficientes (selectivamente) reordenados, utilizando codificación vectorial de longitud variable, codificación de serie-nivel o alguna otra codificación entrópica. Por ejemplo, el codificador realiza la reordenación 1870 tal como sigue, para reordenar información calculada tal como se muestra en la figura 18b y señalizada tal como se muestra en la figura 18c.  
50

En resumen, el codificador reordena coeficientes y entrega los coeficientes a una nueva memoria intermedia de coeficientes (o, directamente, a un codificador entrópico, de manera que el proceso de reordenación no utiliza recursos extra para el almacenamiento en memoria intermedia). El codificador analiza una lista (descrita anteriormente) que indica las posiciones de comienzo y/o las posiciones de finalización de periodos para los cuales serán reordenados los coeficientes. Generalmente, el codificador trata desde el primero de dichos periodos hasta el último de dichos periodos.

Para un periodo, el codificador encuentra el primer coeficiente no procesado aún en la reordenación. Si el coeficiente está en el interior de una región de preámbulo, el codificador entrega el coeficiente y dichos uno o varios coeficientes de preámbulo, en su orden original. De lo contrario, el codificador entrega únicamente el primer coeficiente aún no procesado. A continuación, el codificador marca cualesquiera coeficientes procesados en el periodo, como habiendo sido procesados. El codificador sigue con el primer coeficiente no procesado del siguiente periodo.

Si, para algún periodo, no existen coeficientes no procesados, el codificador simplemente pasa al periodo siguiente.

Después de que el codificador verifica todos los periodos en una iteración desde el primero al último, el codificador repite desde el primer periodo. Finalmente, el codificador procesa la totalidad de los coeficientes en los periodos a reordenar. Cuando los coeficientes en el segmento no están reordenados, el codificador puede simplemente copiar dichos coeficientes a la nueva memoria intermedia de coeficientes (o enviarlos directamente al codificador entrópico en los momentos apropiados).

Alternativamente, el codificador lleva a cabo la reordenación 1870 utilizando algún otro mecanismo. O bien, el codificador lleva a cabo la reordenación 1870 de acuerdo con información de reordenación diferente y/o adicional.

A continuación, el codificador determina 1890 si continuar con el segmento siguiente o finalizar la técnica 1800. Si el codificador continúa, el codificador calcula 1810 la información de reordenación para el segmento siguiente, señala 1820 la información de reordenación, y así sucesivamente.

Si bien las figuras 18a hasta 18c muestran las acciones de calcular información de reordenación, como diferentes y anteriores a las acciones de señalar información de realización, estas acciones pueden, alternativamente, intercarse entre sí o con otras acciones.

#### **D. Ejemplos de técnicas para reordenar coeficientes durante la descodificación.**

La figura 19a muestra una técnica 1900 para la reordenación de los coeficientes espectrales cuantificados durante la descodificación, y las figuras 19b y 19c detallan posibles formas de llevar a cabo ciertas acciones de la técnica 1900. Por ejemplo, un descodificador tal como el mostrado en la figura 12, lleva a cabo la técnica 1900. Alternativamente, otro descodificador lleva a cabo la técnica 1900.

Inicialmente, el descodificador obtiene 1910 información de reordenación para un segmento. Habitualmente, el descodificador lee información complementaria procedente de un flujo de bits, para utilizar en la intercalación/reordenación. Por ejemplo, el codificador obtiene 1910 información de reordenación tal como se muestra en la figura 19b, para reordenar información señalizada tal como se muestra en la figura 18c. Alternativamente, el descodificador obtiene información de reordenación diferente y/o adicional.

Haciendo referencia a la figura 19b, el descodificador analiza sintácticamente 1912 un bit de activación/desactivación para la reordenación a partir del flujo de bits. Por ejemplo, el descodificador lee un sólo bit desde el flujo de bits, donde el único bit indica si utilizar un modo con reordenación de coeficientes o un modo sin reordenación de coeficientes. Alternativamente, la información de activación/desactivación es señalizada y analizada sintácticamente utilizando algún otro mecanismo y/o para alguna duración diferente a un segmento completo.

Cuando se utiliza reordenación de coeficientes (bifurcación "sí" de la decisión 1914), el descodificador analiza sintácticamente 1916 la longitud de periodo a partir del flujo de bits. Cuando se permiten longitudes de periodos no enteras, la longitud del periodo puede representarse con una parte entera y una parte fraccionaria, que son ambas analizadas sintácticamente a partir del flujo de bits. Una longitud de periodo entera (o una parte entera de una longitud de periodo no entera) es representada como un FLC con  $\log_2(\text{longitud\_del\_periodo\_más\_larga})$  bits. Alternativamente, la longitud del periodo es señalizada con otro mecanismo.

Asimismo, el descodificador analiza sintácticamente 1918 el primer periodo para el cual los coeficientes serán reordenados a partir del flujo de bits, lo cual indica aproximadamente la posición de comienzo de la reordenación. El primer periodo reordenado puede representarse en unidades de la longitud del periodo. El primer periodo reordenado es representado, por ejemplo, con un FLC de tres bits. Alternativamente, el primer periodo reordenado es señalizado y analizado sintácticamente con otro mecanismo.

Asimismo, el descodificador analiza sintácticamente 1940 el último periodo para el cual los coeficientes serán reordenados a partir del el flujo de bits. El último período reordenado puede representarse en unidades de la longitud del periodo. El último período reordenado es señalado, por ejemplo, como un FLC con  $\log_2(\text{número\_máximo\_de\_periodos})$  bits, donde el descodificador obtiene el número máximo de periodos a través a partir del número de coeficientes en el segmento y de la longitud del periodo. Alternativamente, el último período reordenado es señalado y analizado sintácticamente con otro mecanismo.

Con la longitud del periodo, el primer período reordenado y el último período reordenado, el descodificador tiene información para llenar una tabla de posiciones de periodos que almacena las posiciones de comienzo y/o posiciones de finalización de los períodos, con propósitos de seguimiento. Por lo tanto, el descodificador puede reproducir la tabla de posiciones de periodos utilizada por un codificador correspondiente.

El descodificador analiza sintácticamente 1902 ajustes de posiciones a partir del flujo de bits. Para los periodos para los cuales se reordenarán los coeficientes, el descodificador analiza sintácticamente información que indica desplazamientos en relación con posiciones iniciales de comienzo y/o de finalización. Por ejemplo, se analiza sintácticamente un valor de ajuste por periodo, y el valor de ajuste es representado como un número de coeficientes. Dicho valor de ajuste puede ser representado como un FLC con  $\log_2(\text{rango\_de\_desplazamiento})$  bits. Alternativamente, el valor de ajuste es señalado y analizado sintácticamente por otro mecanismo.

Con la información del ajuste de posiciones, el descodificador tiene información para ajustar las posiciones de comienzo y/o las posiciones de finalización de los periodos, en la tabla de posiciones de periodos.

Asimismo, el descodificador analiza sintácticamente 1924 un valor de preámbulo. Un valor de preámbulo de algún número de coeficientes es representado como un FLC con  $\log_2(\text{preámbulo\_mayor} + 1)$  bits. Alternativamente, el valor de preámbulo es señalado y analizado sintácticamente por otro mecanismo.

Volviendo a la figura 19a, el descodificador realiza la descodificación entrópica 1930 de la información de coeficientes a partir del flujo de bits, utilizando descodificación vectorial de longitud variable, descodificación de serie-nivel o alguna otra descodificación entrópica. Cuando no se utilizó reordenación en la codificación, el descodificador realiza la descodificación entrópica 1930 de los coeficientes espectrales cuantificados del segmento en su orden original. Por otra parte, cuando se utilizó reordenación en la codificación, el descodificador realiza descodificación entrópica 1930 de los coeficientes espectrales cuantificados reordenados.

Asimismo, el descodificador determina 1960 si se utiliza o no reordenación de coeficientes durante la descodificación. Si se utiliza reordenación de coeficientes durante la descodificación, el descodificador reordena 1970, por lo menos, parte de los coeficientes del segmento sometidos a descodificación entrópica. Por ejemplo, el descodificador lleva a cabo la reordenación 1970 como sigue, para reordenar información recibida tal como se muestra en la figura 19b.

El descodificador genera 1972 una tabla de posiciones de períodos a partir de información de reordenación para el segmento (por ejemplo, longitud de período, primer período reordenado, último período reordenado) y aplica 1974 ajustes de periodos a la tabla. La tabla almacena las posiciones de comienzo y/o las posiciones de finalización de periodos para utilizar en la reordenación. Alternativamente, el descodificador se salta el proceso de generación de la tabla o utiliza alguna otra estructura de tabla.

A continuación, el descodificador reordena 1976 coeficientes utilizando la tabla de posiciones de períodos y el valor de preámbulo. En resumen, el descodificador reordena coeficientes y entrega los coeficientes a una nueva memoria intermedia de coeficientes, invirtiendo la reordenación realizada durante la codificación. (Alternativamente, el descodificador puede reordenar las salidas del descodificador entrópico directamente, de manera que no se utilizan recursos adicionales para el almacenamiento de coeficientes en memoria intermedia.) El descodificador utiliza la tabla de posiciones de periodos (descrita anteriormente), que indica las posiciones de comienzo y/o las posiciones de finalización de periodos para los cuales los coeficientes deben ser reordenados. En general, el descodificador procesa los coeficientes espectrales sometidos a descodificación entrópica, en el orden resultante de la descodificación entrópica. Por ejemplo, en posiciones para el primer período reordenado, el descodificador coloca el primer coeficiente no procesado así como cualesquiera coeficientes no procesados en la región de preámbulo para el primer período reordenado. A continuación, en posiciones para el segundo período reordenado, el descodificador coloca el siguiente coeficiente no procesado así como cualesquiera coeficientes no procesados en la región de preámbulo para el segundo período reordenado. El descodificador repite este proceso de preámbulo para cada uno de los periodos hasta el último período reordenado. A continuación, el descodificador coloca de manera iterativa sucesivos coeficientes no procesados, en posiciones para el primer, segundo, tercer, etc. periodos reordenados, saltando un período reordenado cuando dicho período reordenado se ha llenado. Finalmente, el descodificador procesa la totalidad de los coeficientes en los periodos a reordenar. Cuando los coeficientes en el segmento no están reordenados, el descodificador puede simplemente copiar dichos coeficientes a posiciones correspondientes en la nueva memoria intermedia de coeficientes.

Alternativamente, el descodificador lleva a cabo reordenación 1970 utilizando algún otro mecanismo. Por ejemplo, utilizando la tabla de posiciones de períodos y el valor de preámbulo, el descodificador busca a través de los coeficientes sometidos a descodificación entrópica, seleccionando y entregando coeficientes espectrales para el primer período reordenado. A continuación, el codificador busca a través de los coeficientes sometidos a descodificación entrópica, seleccionando y entregando coeficientes espectrales para el segundo período reordenado, y así sucesivamente, hasta el último período reordenado. O bien, el descodificador lleva a cabo la reordenación 1970 de acuerdo con información de reordenación diferente y/o adicional.

A continuación, el descodificador determina 1990 si continuar con el segmento siguiente o finalizar la técnica 1900. Si el descodificador continúa, el descodificador obtiene 1910 la información de la reordenación para el segmento siguiente, y así sucesivamente.

Si bien las figuras 19a hasta 19c muestran las acciones de obtención de información de reordenación como diferentes y anteriores a las otras acciones de reordenación, alternativamente estas acciones están intercaladas entre sí o con otras acciones.

#### E. Resultados.

En general, la reordenación de los coeficientes espectrales cuantificados mejora la eficiencia de la subsiguiente codificación entrópica para señales periódicas. Por ejemplo, la reordenación agrupa localmente coeficientes que tienen valores similares, haciendo más eficiente la subsiguiente codificación vectorial de longitud variable y/o codificación de serie-nivel.

La reordenación descrita anteriormente es de implementación relativamente simple, y tiene una complejidad computacional reducida. En relación con el uso de memoria, en algunas implementaciones, la única memoria extra requerida por las operaciones de reordenación es una tabla de posiciones de períodos, que es muy pequeña.

La figura 20 muestra los coeficientes espectrales de la figura 15b después de la reordenación de coeficientes. La longitud del período de 56,7. La reordenación comienza en la posición 114 (comenzando el tercer período en el segmento), y la reordenación finaliza en torno a la posición 1021 (finalizando el período 18-ésimo en el segmento). El preámbulo vale tres para los períodos del segmento. Después de la reordenación, los coeficientes hasta aproximadamente la posición 250 son apropiados para codificación vectorial de longitud variable, y los coeficientes a continuación son apropiados para codificación de serie-nivel.

La ganancia de codificación atribuible a la reordenación depende de la periodicidad de la señal. Si una señal es periódica en el dominio temporal, existe a menudo una ganancia significativa reordenando los coeficientes espectrales. De lo contrario, las ganancias de codificación son habitualmente menos significativas o inexistentes. La figura 21 muestra la ganancia de codificación debida a la reordenación por subtrama de un archivo de audio de ejemplo, con una señal periódica. La mayor ganancia para una subtrama está por encima del 40%, y la ganancia promedio para el archivo es de aproximadamente el 11%.

#### V. Utilización selectiva de múltiples modelos entrópicos en codificación/descodificación adaptativa.

En algunas realizaciones, un codificador tal como uno mostrado en la figura 2, 4, ó 6 realiza codificación entrópica adaptativa, en la cual el codificador utiliza selectivamente múltiples modelos entrópicos. Un descodificador correspondiente (tal como uno mostrado en las figuras 3, 5 ó 7) lleva a cabo descodificación entrópica adaptativa, en la cual el descodificador utiliza selectivamente múltiples modelos entrópicos. Las técnicas y herramientas para el uso selectivo de múltiples modelos entrópicos son aplicables en diversos escenarios, en los cuales los valores de símbolo tienen múltiples distribuciones de probabilidad, incluyendo compresión y descompresión, sin pérdidas y con pérdidas, de audio, video, imágenes o cualesquiera otros datos.

#### A. Ejemplo de dominio del problema.

La codificación adaptativa de símbolos se utiliza a menudo para mejorar la eficiencia de la codificación entrópica cuando la distribución de probabilidad para los valores de símbolos varía. La codificación aritmética adaptativa puede utilizar directamente distribuciones de probabilidad diferentes o cambiantes. Para la codificación de longitud variable adaptativa (tal como la codificación de Huffman), se realizan diferentes modelos entrópicos para valores de símbolos, en tablas VLC diferentes o cambiantes.

Con adaptación hacia atrás, la codificación/descodificación se adapta en base a símbolos ya procesados. Con adaptación hacia delante, la información que describe la adaptación es señalizada explícitamente. Por ejemplo, un código de conmutación de tabla es señalizado para indicar una tabla VLC a utilizar para una serie de símbolos.

La adaptación puede conseguirse variando dinámicamente la distribución de probabilidad (o los VLC correspondientes utilizados para codificación/descodificación de longitud variable). O la adaptación puede conseguirse escogiendo entre un conjunto fijo de distribuciones (o tablas VLC correspondientes) de probabilidad diferentes, pre-aprendidas.

- 5 Un inconveniente de utilizar múltiples distribuciones/tablas VLC diferentes, es la memoria necesaria para el codificador y el descodificador, puesto que la memoria utilizada crece linealmente con el número de distribuciones/tablas VLC. Por ejemplo, si se utilizan 16 tablas VLC, entonces se utiliza aproximadamente 16 veces más memoria para las tablas VLC, en el codificador y el descodificador, que con el caso de una única tabla VLC.

10 En resumen, se ha descrito un problema que pueden tratar las técnicas y herramientas para el uso selectivo de múltiples modelos entrópicos. Sin embargo, no es necesario aplicar dichas técnicas herramientas para tratar este problema.

B. Utilización selectiva de múltiples modelos entrópicos.

15 La utilización selectiva de múltiples modelos entrópicos puede reducir significativamente la utilización de recursos para múltiples distribuciones/tablas VLC. Al mismo tiempo, sigue siendo posible conseguir gran parte de la ganancia de codificación asociada con la utilización de múltiples modelos entrópicos. En diversos escenarios comunes, utilizar selectivamente múltiples modelos entrópicos implica elegir entre diferentes distribuciones/tablas VLC para algunos, pero no todos, los valores de símbolo. Más en general, implica elegir entre diferentes distribuciones/tablas VLC que están organizadas jerárquicamente para permitir más adaptatividad para algunos valores de símbolo y menos adaptatividad para otros valores de símbolo.

20 Supóngase un conjunto de valores de símbolo que incluye ciertos valores de símbolo más probables y ciertos valores de símbolo menos probables, de acuerdo con alguna prueba. Para reducir la memoria utilizada para las distribuciones/tablas, un codificador y un descodificador utilizan múltiples distribuciones/tablas para los valores de símbolo más probables, pero los valores de símbolo menos probables no están representados en múltiples distribuciones/tablas. Esto reduce la memoria utilizada para las múltiples distribuciones/tablas, con una penalización despreciable sobre la ganancia de codificación. (En muchas situaciones, una fracción relativamente pequeña de los valores de símbolo son responsables de un gran porcentaje de una distribución de probabilidad.). En particular, si se ve el modelo entrópico como siendo condicional para un estado dado de adaptación, existe una distribución diferente para los valores de símbolo más probables en los diferentes estados respectivos. Sin embargo, la distribución relativa para los valores de símbolo menos probables es idéntica en los diferentes estados.

30 Para un conjunto de 256 valores de símbolo, si la mayor parte del tiempo se utilizan 32 de los valores de símbolo, un codificador y un descodificador pueden conmutar entre 6 tablas VLC para los 32 valores de símbolo, donde cada una de las 6 tablas VLC incluye asimismo un código de escape para conmutar a una sola tabla VLC para los otros 224 valores de símbolo.

35 O bien, supóngase que para el conjunto de 256 valores de símbolo, 7 de los valores de símbolo son utilizados la mayor parte del tiempo, con 21 de los valores de símbolo utilizados ocasionalmente, y el resto de los valores de símbolo utilizados solamente raras veces. El codificador y el descodificador pueden conmutar entre 11 tablas VLC para los 7 valores de símbolo más comunes, donde cada una de las 11 tablas VLC incluye un código de escape para conmutar a 2 tablas VLC para los 21 siguientes valores de símbolo más comunes. (El código de escape puede estar seguido de la información de selección de tabla, para adaptación hacia delante.) Cada una de las 2 tablas VLC para los 21 valores de símbolo incluye un código de escape para conmutar a una tabla VLC para el resto de los valores de símbolo.

45 La figura 22 muestra un ejemplo que es más complejo, en términos de organización jerárquica de los modelos entrópicos/estados (por ejemplo, distribuciones, tablas VLC). Un codificador y un descodificador utilizan 8 modelos entrópicos para los valores de símbolo B, F, H e I, donde cada uno de los 8 modelos entrópicos incorpora asimismo dos puntos de conmutación. Por ejemplo, si el codificador y el descodificador utilizan distribuciones de probabilidad para los modelos entrópicos, un punto de conmutación es un valor de probabilidad especial de conmutación, en una distribución. Si el codificador y el descodificador utilizan tablas VLC para los modelos entrópicos, un punto de conmutación es un código de escape u otro VLC especial. En los 8 modelos entrópicos, el primer punto de conmutación es para conmutar a modelos entrópicos para los valores de símbolo A y C, y el segundo punto de conmutación es para conmutar a modelos entrópicos para los valores de símbolo D, E, G, J y K.

50 El codificador y el descodificador utilizan 3 modelos entrópicos para los valores de símbolo A y C. El codificador y el descodificador utilizan 4 modelos entrópicos para los valores de símbolo E, J y K, donde cada uno de los 4 modelos entrópicos incorporan asimismo un punto de conmutación. Este punto de conmutación es para conmutar a un modelo entrópico para los valores de símbolo D y G.

En la figura 22, un subconjunto de los valores de símbolo tiene menos modelos entrópicos asociados que su superconjunto. Esto es consistente con muchos escenarios comunes en los que se permite más adaptatividad para valores de símbolo más probables, y se permite menos adaptatividad para valores de símbolo menos probables. Sin embargo, alternativamente un subconjunto puede tener más modelos entrópicos asociados que su superconjunto.

5 La selección entre múltiples modelos entrópicos puede ser a través de un mecanismo adaptativo hacia atrás, o de un mecanismo adaptativo hacia delante. Los propios múltiples modelos entrópicos pueden ser fijos y pre-aprendidos, o pueden cambiar dinámicamente. Los modelos entrópicos pueden ser aplicados en diversos esquemas de codificación y descodificación entrópica. La codificación y descodificación aritmética puede utilizar selectivamente múltiples distribuciones de probabilidad para algunos valores de símbolo pero no para todos. O la codificación y descodificación de longitud variable puede utilizar múltiples tablas VLC para parte, pero no la totalidad, de los valores de símbolo.

1. Ajuste de distribuciones para estados.

15 Para que un codificador o un descodificador utilice selectivamente múltiples modelos entrópicos para algunos valores de símbolo (pero no para todos los valores de símbolo), los múltiples modelos entrópicos se ajustan en consecuencia. El siguiente análisis ilustra ajustes para probabilidades de distribución reales para un conjunto de estados, haciendo referencia a un ejemplo simple.

Supóngase que existen N estados etiquetados como  $S(j) = S(0), S(1), \dots, S(N-1)$ , para adaptar una distribución de M valores de símbolo, etiquetados como  $X(i) = X(0), X(1), \dots, X(M-1)$ .

20  $P_S$  indica distribuciones de probabilidad para los estados, siendo  $P_{S(j)}$  la probabilidad de que el estado sea  $S(j)$ .  $P_{S(j),X}$  indica la distribución de probabilidad para los valores de símbolo estado el estado  $S(j)$ , siendo  $P_{S(j),X(i)}$  la probabilidad de que el símbolo tenga un valor  $X(i)$  estando en el estado  $S(j)$ . Entre los M valores de símbolo, se designan L valores de símbolo como más probables, y se designan M - L valores de símbolo como menos probables. El conjunto de los L valores de símbolo más probables es el conjunto Q, y el conjunto de los M - L valores de símbolo menos probables es el conjunto R.

25 La designación de valores de símbolo más probables frente a otros menos probables depende de la implementación y es flexible, si bien una designación apropiada conduce una modificación más eficiente. No es necesario que  $P_{S(j),X(q)} > P_{S(j),X(r)}$  para todos los estados  $S(j)$ , donde  $X(q)$  indica un valor de símbolo en Q y  $X(r)$  indica un valor de símbolo en R. En otras palabras, no se requiere que un valor de símbolo "más probable" dado, tenga una probabilidad mayor que un valor de símbolo "menos probable" dado, en todos los estados.

30 Una distribución revisada  $P'_{S(j),X}$  para el estado  $S(j)$  aproxima la distribución real de valores de símbolo  $P_{S(j),X}$  para el estado  $S(j)$ .  $P'_{S(j),X}$  aproxima  $P_{S(j),X}$  de tal modo que: (1) la distribución condicional  $P'_{S(j),X(i),R}$  para los varones de símbolo  $X(i)$  en el conjunto R es la misma para todos los  $S(j)$ , pero (2) la distribución para valores de símbolo en el conjunto Q no cambia para cualquier  $S(j)$  dado ( $P'_{S(j),X(i)} = P'_{S(j),X(i)}$  para los valores de símbolo  $X(i)$  en el conjunto Q).

35 Supóngase que  $N = 3$  y  $M = 6$ . El conjunto de los estados es  $N = \{S(0), S(1), S(2)\}$ , y el conjunto de los valores de símbolo es  $M = \{X(0), X(1), X(2), X(3), X(4)\}$ .

Asimismo, supóngase que las probabilidades de los estados son  $P_{S(0)} = 0,5$ ,  $P_{S(1)} = 0,2$ ,  $P_{S(2)} = 0,3$  tal como se muestra la tabla 3. Por lo tanto, la probabilidad de estar en el estado 0 es del 50%, la probabilidad de estar en el estado 1 es del 20%, y la probabilidad de estar en el estado 2 es del 30%.

Tabla 3. Probabilidades de estados.

$P_{S(0)}$	$P_{S(1)}$	$P_{S(2)}$
0,5	0,2	0,3

40 La tabla 4 muestra las distribuciones de probabilidad reales  $P_{S(j),X(i)}$  para los valores de símbolo en cada uno de los estados.

Tabla 4. Distribuciones de probabilidad reales para valores de símbolo en los estados.

	X(0)	X(1)	X(2)	X(3)	X(4)
$P_{S(0),X(i)}$	0,09	0,4	0,04	0,4	0,07
$P_{S(1),X(i)}$	0,055	0,7	0,03	0,2	0,015
$P_{S(2),X(i)}$	0,165	0,1	0,09	0,6	0,045

Como un umbral arbitrario, supóngase que un valor de símbolo  $X(i)$  pertenece al conjunto más probable  $Q$  si, para cualquiera de los estados, la probabilidad del valor de símbolo en el estado, multiplicada por la probabilidad de estar en ese estado, es mayor que 0,1. Es decir, si  $P_{S(j),X(i)} * P_{S(j)} > 0,1$  para cualquier  $S(j)$  para un  $X(i)$  dado, entonces el valor de símbolo  $X(i)$  está en el conjunto  $Q$ . De lo contrario, el valor de símbolo  $X(i)$  está en el conjunto  $R$ . Para las distribuciones en la tabla 4,  $L = 2$ ,  $Q = \{X(1), X(3)\}$  y  $R = \{X(0), X(2), X(4)\}$  (Obsérvese que incluso aunque  $P_{S(2),X(0)} > P_{S(2),X(1)}$ , el valor de símbolo  $X(1)$  se designa como un valor de símbolo más probable, mientras que el valor de símbolo  $X(0)$  se designa como un valor de símbolo menos probable. En el estado  $S(1)$ ,  $X(1)$  tiene una probabilidad elevada.) Alternativamente, el valor umbral y/o la prueba son diferentes. Por ejemplo, el umbral se determina en términos del porcentaje de valores de símbolo, o la prueba requiere una probabilidad elevada en múltiples estados diferentes. En general, para una restricción dada sobre el tamaño de conjuntos  $Q$  y  $R$ , puede encontrarse una división óptima observando la entropía relativa entre las distribuciones real y aproximada. (En general, tal como se utiliza en el presente documento, el término "óptima" describe una solución que satisface mejor que otras soluciones un conjunto de criterios, de acuerdo con alguna parametrización o modelización, la cual puede, o no, ser óptima en términos absolutos dependiendo de circunstancias, y el término "optimizar" se utiliza para indicar el proceso de encontrar una solución de este tipo.)

En la aproximación,  $P'_{S(j),X(i)} = P_{S(j),X(i)}$  para valores de símbolo  $X(i)$  en el conjunto  $Q$ . La distribución para un estado  $S(j)$  no se modifica para valores de símbolo en el conjunto  $Q$ . Sin embargo, para valores de símbolo  $X(i)$  en el conjunto  $R$ , la distribución aproximada es diferente. Para empezar, se calculan las distribuciones condicionales reales  $P_{S(j),X(i),R}$  para los valores de símbolo en el conjunto  $R$ . Para los valores de símbolo en el conjunto  $R$ , en la tabla 5 se proporcionan las distribuciones condicionales reales eliminando las contribuciones de los valores de símbolo  $X(1)$ ,  $X(3)$  en el conjunto  $Q$ , y ponderando solamente las contribuciones a partir de  $X(0)$ ,  $X(2)$ ,  $X(4)$ .  $P_{S(0),X(0),R}$  es  $0,09 / (0,09 + 0,04 + 0,07) = 0,45$ , y  $P_{S(0),X(1),R}$  es  $0,04 / (0,09 + 0,04 + 0,07) = 0,2$ .

Tabla 5. Distribuciones condiciones reales para valores de símbolo en el conjunto  $R$ .

	X(0)	X(2)	X(4)
$P_{S(0),X(i),R}$	0,45	0,2	0,35
$P_{S(1),X(i),R}$	0,55	0,3	0,15
$P_{S(2),X(i),R}$	0,55	0,3	0,15

A continuación, la distribución condicional aproximada  $P'_{S(j),X(i),R}$  se calcula como:

$$P'_{S(j),X(i),R} = \sum_{S(j)=S(0)}^{S(N-1)} P_{S(j)} * P_{S(j),X(i),R} \tag{1}$$

Es decir, la distribución condicional aproximada en el conjunto  $R$  es la media ponderada (mediante  $P_{S(j)}$ ) de la distribución condicional real  $P_{S(j),X(i),R}$  sobre los  $N$  estados). Para los valores de las tablas 4 y 5, en la tabla 6 se muestra la distribución condicional aproximada  $P'_{S(j),X(i),R}$  en el conjunto  $R$ . Para  $X(0)$ ,  $P'_{S(j),X(0),R}$  es  $(0,5 * 0,45) + (0,2 * 0,55) + (0,3 * 0,55) = 0,5$ .

Tabla 6. Distribución condicional aproximada para valores de símbolo en el conjunto  $R$ .

	X(0)	X(2)	X(4)
$P'_{S(j),X(i),R}$	0,5	0,25	0,25

La distribución final aproximada para cada estado  $S(j)$  es:

$$P'_{S(j),X(i)} = \begin{cases} P'_{S(j),X(i),R} * \sum_{X(i) \in R} P_{S(j),X(i)} & \text{si } X(i) \in R \\ P_{S(j),X(i)} & \text{si } X(i) \in Q \end{cases} \quad (2).$$

De este modo, para los valores de símbolo en el conjunto Q, el valor de probabilidad real en el estado S(j) se utiliza en la distribución aproximada para el estado S(j). Para un valor de símbolo en el conjunto R, la distribución de probabilidad condicional aproximada  $P'_{S(j),X(i),R}$  para el valor de símbolo, se multiplica por la suma de las probabilidades reales para los valores de símbolo en el conjunto R para el estado S(j). Para el valor de símbolo X(0) y el estado S(0),  $P'_{S(0),X(0)}$  es  $0,5 * (0,09 + 0,04 + 0,07) = 0,1$  Para los otros valores en las tablas 4 y 6, las distribuciones de probabilidad finales aproximadas para los estados S(j) se proporcionan en la tabla 7.

Tabla 7. Distribuciones finales aproximadas para valores de símbolo en los estados.

	X(0)	X(1)	X(2)	X(3)	X(4)
$P_{S(0),X(i)}$	0,1	0,4	0,05	0,4	0,05
$P_{S(1),X(i)}$	0,05	0,7	0,025	0,2	0,025
$P_{S(2),X(i)}$	0,15	0,1	0,075	0,6	0,075

10 Básicamente, comparando la tabla 7 con la tabla 4, las distribuciones no han cambiado para los valores de símbolo más probables X(1), X(3), y las distribuciones han cambiado para los valores de símbolo menos probables X(0), X(2), X(4), para reforzar la condición de que la probabilidad relativa para valores de símbolo en el interior del conjunto R es la misma entre estados. En concreto, en cada estado en la tabla en 7, X(0) es el doble del probable que X(2), y X(0) el doble de probable que X(4).

15 Para el caso general, comenzando con N estados para M valores de símbolo, el número de estados para parte de los valores de símbolo (conjunto R) puede reducirse agrupando las N distribuciones condicionales para el conjunto R, en P distribuciones, donde  $P < N$ . A continuación, este procedimiento puede repetirse para algún otro subconjunto de los M valores de símbolo. Asimismo, esto puede repetirse de forma recursiva sobre las P distribuciones agrupadas del conjunto R, donde el conjunto R tiene |R| valores de símbolo (|R| representa el número cardinal o número de elementos del conjunto R) con P estados. Esto impone restricciones sobre los N estados (o distribuciones, o grupos) para los M valores de símbolo. Estas restricciones pueden ser aplicadas después de que han sido fijados los N estados para los M valores de símbolo, o para una mayor optimización pueden ser aplicadas durante la propia fase de aprendizaje. El aprendizaje comenzará con un gran número de distribuciones para los M valores de símbolo, y tendrá como resultado N distribuciones agrupadas, tales que satisfarán las restricciones extra sobre las distribuciones condicionales.

**2. Ejemplos de tablas VLC.**

Las distribuciones aproximadas para valores de símbolo en diferentes estados pueden ser utilizadas en diversos tipos de codificación y decodificación entrópica adaptativa, incluyendo codificación y decodificación de Huffman, y otras codificaciones y decodificaciones de longitud variable.

30 Una tabla de códigos de Huffman puede verse como un árbol, en el que cada hoja del árbol corresponde a un valor de símbolo. La rama izquierda del árbol tiene una asociación con un valor binario (por ejemplo, 0), y la rama derecha del árbol tiene una asociación con el valor binario opuesto (por ejemplo, 1). Los árboles mostrados en la figura 23 corresponden a las distribuciones aproximadas mostradas en la tabla 7.

35 En la figura 23, las partes de trazos de los árboles respectivos son para los valores de símbolo en el conjunto R, y las otras partes de los árboles son para los valores de símbolo en el conjunto Q. En las distribuciones aproximadas mostradas en la tabla 7, la distribución condicional de los valores de símbolo del conjunto R es la misma independientemente del estado, de manera que cada uno de los árboles en la figura 23 puede tener una rama común e idéntica para los valores de símbolo en el conjunto R. La ubicación de la rama idéntica común, puede ser en cualquier posición de un árbol, dependiendo, en general, de cómo el agregado de las probabilidades de los valores de símbolo representados en la rama común se compara con las probabilidades de los otros valores de símbolo para el árbol. De este modo, la rama común podría estar más arriba o más abajo, en uno u otro árbol.

45 Para cualquier árbol/estado dado en la figura 23, los VLCs para todos los valores de símbolo en el conjunto R tienen el mismo prefijo, que se indica mediante la posición de la rama de un árbol. Además, independientemente del estado en la figura 23, cada valor de símbolo del conjunto R tienen un sufijo común, que se indica mediante la rama idéntica común. Para los árboles de la figura 23, los ejemplos de códigos de Huffman son los siguientes.

Tabla 8. Ejemplos de códigos y tablas de Huffman.

	Código de Huffman para S(0)	Código de Huffman para S(1)	Código de Huffman para S(2)
X(0)	<u>11</u> 0	<u>11</u> 0	<u>10</u> 0
X(1)	0	0	11
X(2)	<u>11</u> 10	<u>11</u> 10	<u>10</u> 10
X(3)	10	10	0
X(4)	<u>11</u> 11	<u>11</u> 11	<u>10</u> 11

5 La misma tabla puede ser utilizada para estados S(0) y S(1). En los estados S(0) y S(1), el prefijo común (que se muestra subrayado) para valores de símbolo en el conjunto R es "11", independientemente del valor de símbolo en el conjunto R. En el estado S(2), el prefijo común (que se muestra subrayado) para valores de símbolo en el conjunto R es "10". En los estados S(0), S(1), y S(2), los sufijos (que se muestran en negrita) para los valores de símbolos respectivos son los mismos. (El sufijo para X(0) es "0," el sufijo para X(1) es "10," y el sufijo para X(2) es "11.")

10 En este caso, los códigos de Huffman para las distribuciones aproximadas facilitan, y pueden implementarse con, la codificación/descodificación de dos etapas para valores de símbolo en el conjunto R. Los códigos mostrados en la tabla 8 pueden además dividirse, tal como se muestra en las tablas 9 y 10.

Tabla 9. Tablas de códigos de la primera etapa, para los estados respectivos.

	Código de Huffman para S(0)	Código de Huffman para S(1)	Código de Huffman para S(2)
X(1)	0	0	11
X(3)	10	10	0
X(0), X(2), X(4)	11	11	10

Tabla 10. Tablas de códigos de la segunda etapa, para la totalidad de los estados.

	Código de Huffman para S(0), S(1), y S(2)
X(0)	0
X(2)	10
X(4)	11

15 Para un símbolo que tenga un valor en el conjunto R, el codificador codifica en primer lugar un código de escape que representa la totalidad de los valores de símbolo en el conjunto R. Esto señala una conmutación desde una primera tabla de códigos para valores de símbolo en el conjunto Q para un estado específico, a una segunda tabla de códigos para valores de símbolo en el conjunto R a través de todos los estados. A continuación, el codificador  
 20 codifica el código apropiado a partir de la segunda tabla de códigos.

25 En una organización jerárquica más compleja de las tablas de códigos de Huffman, las tablas de códigos de Huffman pueden incluir múltiples ramas comunes, correspondiendo cada rama común a una sola distribución condicional para un diferente subconjunto de valores de símbolo. En una implementación en dos etapas, las tablas de códigos de Huffman de la primera etapa pueden incluir múltiples códigos de escape, uno para cada una de las múltiples ramas comunes.

Más en general, las tablas de códigos de Huffman pueden organizarse en una jerarquía arbitraria, con códigos de escape (y, posiblemente, otra información de selección) utilizados para conmutar a otra tabla de códigos de Huffman, o conjunto de tablas de códigos de Huffman.

30 En una tabla concreta, un código de escape puede ser utilizado, asimismo, para conmutar a un esquema de codificación/descodificación de longitud fija, para ciertos valores de símbolo (en lugar de conmutar a otra tabla).

Alternativamente, se construyen otros tipos de tablas VLC que no siguen las reglas de los códigos de Huffman. Por ejemplo, una sola tabla VLC asocia VLCs con valores de símbolo en el conjunto R para todo un grupo de estados, y múltiples tablas VLC (una tabla por cada estado del grupo) asocia VLCs con valores de símbolo en el conjunto Q.

5 Además, si bien los ejemplos precedentes ilustran tablas de códigos fijas, pre-aprendidas, alternativamente, las tablas de código varían dinámicamente sus códigos en función de los valores de símbolo que han sido procesados. Para dichas tablas que varían dinámicamente, el codificador y el decodificador pueden seguir utilizando selectivamente múltiples tablas de códigos para algunos valores de símbolo, y una sola tabla de códigos para otros valores de símbolo.

10 En general, si existen N estados para M valores de símbolo, entonces existen N tablas VLC, o N árboles si se utilizan códigos de Huffman. Si existen L subconjuntos disjuntos de los M valores de símbolos, cada uno de los L subconjuntos con  $P_l$  estados, para  $l = 0, 1, \dots, L-1$ , y  $P_l < N$  para todo l, entonces cada uno de los N árboles tendrá L ramas (etiquetadas  $b_0, b_1, \dots, b_{L-1}$ ), escogiéndose cada rama  $b_l$  a partir de una de las ramas comunes  $P_l$  disponibles para dicho subconjunto l. Además, si cualquiera de los subconjuntos L es dividido de nuevo en subconjuntos recursivamente, teniendo cada subconjunto menos estados que su conjunto padre, puede decirse lo mismo sobre las ramas que se ramifican desde las  $P_l$  ramas.

**3. Ejemplos de distribuciones para codificación/descodificación aritmética.**

En otros codificadores/descodificadores se utilizan distribuciones aproximadas en la codificación/descodificación aritmética. La codificación aritmética implica, en general, representar una serie de símbolos como un único número en el interior de un rango dado. Habitualmente, el número es un número fraccionario entre 0 y 1. Un símbolo es codificado colocándolo en parte de un rango, donde el rango es dividido dependiendo de la distribución de probabilidad de los valores de símbolo.

25 Para su utilización en codificación y descodificación aritmética, las distribuciones aproximadas mostradas en la tabla 7 podrían dividirse en la tabla 6 y la tabla 11. El valor de conmutación en la tabla 11 para X(0), X(2) y X(4) indica un cambio desde uno de los estados/distribuciones mostrados en la tabla 11 al estado/distribución mostrado en la tabla 6.

Tabla 11. Distribuciones aproximadas con valores de símbolo en Q combinado.

	X(1)	X(3)	X(0), X(2), X(4)
$P_{S(0),X(i)}$	0,4	0,4	0,2
$P_{S(1),X(i)}$	0,7	0,2	0,1
$P_{S(2),X(i)}$	0,1	0,6	0,3

30 Si bien el ejemplo anterior ilustra distribuciones fijas, pre-aprendidas, alternativamente, las distribuciones varían dinámicamente dependiendo de los valores de símbolo que han sido procesados. Para dichas distribuciones que varían dinámicamente, el codificador y el decodificador pueden seguir utilizando selectivamente múltiples distribuciones para algunos valores de símbolos, y una única distribución para otros valores de símbolo.

**4. Ejemplo de aprendizaje para determinar modelos entrópicos.**

35 Cuando un codificador y un decodificador utilizan selectivamente múltiples modelos entrópicos para símbolos, los modelos entrópicos dependen finalmente de información de la distribución de probabilidad para los símbolos. En algunas implementaciones, una herramienta tal como un codificador, o soporte lógico de análisis estadístico, utiliza el siguiente enfoque para determinar estados y distribuciones de probabilidad para modelos entrópicos.

40 La figura 24 muestra una técnica 2400 de dos etapas para agrupar distribuciones de probabilidad en estados, para un esquema de codificación/descodificación de múltiples modelos entrópicos. La técnica 2400 trata distribuciones de probabilidad de valores de símbolo con vectores de aprendizaje, y los vectores de aprendizaje son agrupados en grupos, de manera similar a los enfoques de agrupamiento utilizados para los esquemas de cuantificación vectorial.

45 Para comenzar, la herramienta obtiene 2410 distribuciones de probabilidad reales para vectores de aprendizaje. Los vectores de aprendizaje proceden de un conjunto de aprendizaje de fuentes representativas. Por ejemplo, para codificación/descodificación de audio, la distribución de probabilidad de valores de símbolo en una subtrama representa un vector de aprendizaje. Para codificación/descodificación de audio general, el conjunto de aprendizaje incluye múltiples fuentes de audio, de tal modo que se obtienen distribuciones de probabilidad para múltiples

subtramas de las diferentes fuentes de audio. Pueden obtenerse vectores de prueba a partir de aprendizaje, a diversas velocidades de bits y/o configuraciones de calidad.

5 A continuación, la herramienta agrupa 2420 los vectores de aprendizaje utilizando una primera métrica de costos. Por ejemplo, la primera métrica de costos es el error cuadrático medio ("MSE", mean squared error). El propio agrupamiento puede utilizar una variación del algoritmo de Lloyd generalizado ("GLA", generalized Lloyd algorithm), que se explica haciendo referencia a la figura 25, o puede utilizar algún otro mecanismo. Básicamente, en la variación GLA, la herramienta agrupa iterativamente vectores de aprendizaje en un número dado de grupos, iterando entre encontrar un codificador óptimo para un descodificador dado, y encontrar un descodificador óptimo para un codificador dado. Después de cierto número de iteraciones, la herramienta encuentra un conjunto de grupos tal que se minimiza la primera métrica de costos.

15 A continuación, la herramienta refina 2430 los grupos, utilizando una segunda métrica de costos. Por ejemplo, la segunda métrica de costos es una métrica entrópica relativa. La distancia Itakura-Saito es una manera de medir la entropía relativa entre dos distribuciones de probabilidad. En el refinamiento 2430, partes de la lógica de agrupamiento pueden ser iguales o diferentes respecto de partes de la lógica de agrupamiento utilizadas en la primera métrica de costos.

20 Por lo tanto, de acuerdo con la figura 24, la herramienta utiliza un proceso de aprendizaje en dos etapas. En la primera etapa, la herramienta utiliza la primera métrica de costos (por ejemplo, MSE) para obtener grupos de funciones de masa de probabilidad ("PMF", probability mass function) aproximadas, para las distribuciones. En una segunda etapa, la herramienta utiliza la segunda métrica de costos (por ejemplo, la distancia Itakura-Saito) para seguir refinando los grupos PMF. El MSE es relativamente simple de calcular, pero no modeliza la entropía como la métrica entrópica relativa con propósitos de codificación/descodificación. Por otra parte, la entropía relativa es una métrica eficaz para refinar grupos, pero puede tener como resultado un agrupamiento no óptimo cuando es la única métrica utilizada. En muchos casos, el aprendizaje en dos etapas es no solamente más rápido en términos de complejidad (puesto que la entropía relativa es más compleja de calcular), sino que asimismo tiene como resultado grupos mejores para aplicaciones de codificación/descodificación.

Alternativamente, una herramienta utiliza otro enfoque para determinar estados y distribuciones de probabilidad. Por ejemplo, la herramienta utiliza otra métrica diferente al MSE o a la entropía relativa, para la primera o la segunda métrica de costos. O la herramienta utiliza una sola métrica de costos en un proceso de una sola capa.

30 La figura 25 muestra una técnica 2500 para agrupar vectores de aprendizaje, de acuerdo con una variación de GLA. Tal como en la figura 24, la técnica 2500 trata distribuciones de probabilidad de valores de símbolo como vectores de aprendizaje, y los vectores de aprendizaje son agrupados en grupos.

35 Para comenzar, la herramienta calcula 2510 un solo grupo a partir de vectores de aprendizaje. Para codificación/descodificación de audio en general, por ejemplo, los vectores de aprendizaje son distribuciones de probabilidad para subtramas procedentes de diferentes fuentes de audio, tales como archivos de audio codificados a diferentes velocidades de bits y/o configuraciones de calidad. El número de vectores de aprendizaje obtenido depende de la implementación. En una implementación, la herramienta obtiene aproximadamente 100 veces más vectores de aprendizaje que grupos finales calculados. El grupo único es el centroide de los vectores de aprendizaje, calculado promediando los vectores de aprendizaje, o alguna otra combinación de los vectores de aprendizaje.

40 A continuación, la herramienta divide 2520 en el grupo único en múltiples grupos. Por ejemplo, la herramienta utiliza análisis de componentes principales para dividir el grupo único en dos grupos; uno es el grupo original y el otro es el grupo original más una constante, que depende de la implementación, multiplicada por el componente principal (por ejemplo, el otro es un grupo que está a cierto desplazamiento a lo largo de la dirección del componente principal). Alternativamente, la herramienta utiliza algún otro análisis para dividir el grupo en múltiples grupos.

45 La herramienta clasifica 2530 los vectores de aprendizaje entre los múltiples grupos actuales, de acuerdo con alguna métrica de costos. Por ejemplo, la métrica de costos es MSE, entropía relativa o alguna otra métrica. El MSE de un vector de aprendizaje, respecto de un grupo, indica la distancia euclídea entre los puntos de la distribución de probabilidad del vector de aprendizaje, y los puntos correspondientes del grupo. La entropía relativa entre un vector de aprendizaje y un grupo puede proporcionar la diferencia entre un vector de aprendizaje y un grupo, como sigue:

$$-\sum_k \text{vector\_de\_aprendizaje} * \log_2(\text{grupo}_k) \quad (3),$$

50 donde k indica puntos en el vector de aprendizaje y el grupo. De manera menos formal, la entropía relativa indica una penalización en la velocidad de bits debida al desajuste entre el vector de aprendizaje y el grupo. La herramienta clasifica un vector de aprendizaje con el grupo frente al cual el vector de aprendizaje tiene el MSE menor, la entropía relativa menor, etc.

La herramienta recalcula 2540 los grupos actuales a partir de los vectores de aprendizaje clasificados. Por ejemplo, para el grupo actual, la herramienta calcula el centroide de los vectores de aprendizaje clasificados para dicho grupo. Alternativamente, la herramienta recalcula cada grupo actual, como alguna otra combinación de los vectores de aprendizaje clasificados para dicho grupo.

5 La herramienta determina 2545 si los grupos se han estabilizado. Por ejemplo, la herramienta verifica si el cambio entre los grupos anteriores y los siguientes en el recálculo 2540 satisface algún criterio. Un criterio es que en el recálculo 2540 los grupos no se hayan desplazado en más de cierta cantidad umbral, en donde la cantidad umbral depende de la implementación. Alternativamente, la herramienta considera criterios diferentes y/o adicionales. Si los grupos no se han estabilizado, la herramienta clasifica 2530 los vectores de aprendizaje entre los grupos actuales (recalculados (2540)), de acuerdo con la métrica de costos.

15 Cuando los grupos actuales se han estabilizado, la herramienta determina 2550 si existen grupos suficientes. En general, el número deseado de grupos puede ajustarse como un compromiso entre la utilización de memoria y al rendimiento de la codificación. Tener más grupos tiende a conducir a más estados y adaptatividad en los modelos entrópicos, al costo de una utilización incrementada de memoria para almacenar las distribuciones, tablas VLC, etc. Cuando se utiliza adaptación hacia delante, tener más grupos significa asimismo que se señala más información complementaria (por ejemplo, para indicar distribuciones, tablas, etc.). Por contraste, tener menos grupos tiende a incrementar el desajuste entre los vectores de aprendizaje y los grupos finales, lo que habitualmente indica un desajuste incrementado entre los modelos entrópicos y las distribuciones reales de los valores de símbolo durante la codificación.

20 Si no se ha alcanzado el número deseado de grupos, la herramienta divide 2560 parte o la totalidad de los grupos actuales. Por ejemplo, la herramienta utiliza análisis de componentes principales o algún otro análisis, para dividir un grupo en dos grupos. Supóngase que la herramienta busca G grupos finales, y actualmente tiene F grupos actuales, donde  $F < G$ . Dividir cada uno de los F grupos actuales tendría como resultado demasiados grupos, y la herramienta puede dividir cada uno de los  $G - F$  grupos actuales superiores (por ejemplo, "superiores" en términos de cuántos vectores de aprendizaje están clasificados en los grupos actuales) en dos grupos. O la herramienta puede dividir simplemente el grupo superior en cada interacción, o utilizar alguna otra regla para la división. A continuación, la herramienta clasifica 2530 los vectores de aprendizaje entre grupos actuales (divididos (2560)), de acuerdo con la métrica de costos.

30 Cuando los grupos actuales se han estabilizado y se ha alcanzado el número deseado de grupos, la técnica 1500 finaliza. La clasificación 2530, el recálculo 2540 y la división 2560 constituyen esencialmente una iteración de la variante GLA, y durante las iteraciones la métrica de costos puede reducirse.

35 La técnica 2500 de la figura 25 puede incorporarse a la técnica 2400 de la figura 24, tal como sigue. La herramienta lleva a cabo la técnica 2500 de la figura 25 utilizando MSE como la métrica de costos, hasta que se alcanza el número deseado de grupos. En ese momento, la herramienta lleva a cabo de manera iterativa la clasificación 2530, el recálculo 2540, y verifica 2545 la estabilidad utilizando entropía relativa como métrica de costos, hasta que los grupos se estabilizan/no se desplazan en más de cierta cantidad umbral.

40 Las técnicas 2400, 2500 pueden ser utilizadas para producir grupos finales con distribuciones de probabilidad que se aproximan a las distribuciones reales, pero pueden tener la misma distribución condicional para ciertos valores de símbolo. En términos del marco analítico de la sección V.B.1, las técnicas 2400, 2500 pueden ser utilizadas para producir distribuciones de probabilidad aproximadas, tales como las mostradas en la tabla 7 mediante añadir, en las operaciones de clasificación y agrupamiento, la restricción de que la distribución condicional para los valores de símbolo en un conjunto R sea la misma para todos los grupos/estados ( $P'_{S(i),X(i),R}$  es la misma para todos los estados  $S(i)$ ). Esencialmente, aquellas dimensiones de los grupos que corresponden a valores de símbolo en el conjunto R se restringen tal como se muestra en las ecuaciones (1) y (2). En el análisis, la probabilidad  $P_{S(i)}$  de estar en un estado dado se indica mediante el número de vectores de aprendizaje clasificados en el grupo para dicho estado. Otra limitación es que las dimensiones de cada uno de los grupos sume 1.

50 Haciendo referencia a la figura 25, después del recálculo 2540 de los grupos actuales, pueden imponerse una o más restricciones a la distribución condicional. En general, supóngase que existen N estados para M valores de símbolo, y que existen L subconjuntos de M valores de símbolo, cada uno de los L subconjuntos con  $P_i$  estados,  $P_i < N$ ,  $i = 0, 1, \dots, L-1$ , y E<sub>i</sub> elementos. La totalidad de los valores de símbolo en el interior de uno concreto de los L subconjuntos, puede agruparse en un valor de símbolo (escape/conmutación) común. Existirán L de dichos valores de símbolo de escape/conmutación. A continuación, el aprendizaje pasa a encontrar los N grupos (o distribuciones) para los  $M - (E_0 + E_1 + \dots + E_{L-1}) + L$  valores de símbolo (restando los E<sub>i</sub> elementos en los L subconjuntos, y añadiendo L elementos para los valores de símbolo de escape/conmutación). A continuación, para cada uno de los L subconjuntos de los M valores de símbolo, se calculan una o varias distribuciones condicionales en el interior del conjunto. El aprendizaje se repite sobre cada uno de los L subconjuntos para encontrar  $P_i$  grupos,  $i = 0, 1, \dots, L-1$ , para cada uno de estos subconjuntos. Los vectores de aprendizaje para esto, serán la distribución o distribuciones condicionales en el interior de los L subconjuntos, respectivamente. Si cualquiera de los L subconjuntos es subdividido adicionalmente,

el procedimiento puede repetirse recursivamente para dicho subconjunto subdividido  $I$ , puesto que actualmente existen  $P_i$  estados para  $E_i$  valores de símbolo.

5 En cuanto a la designación de qué valores de símbolo están en los conjuntos  $Q$  y  $R$ , inicialmente esto se basa en la distribución de probabilidad del único grupo inicial. Por consiguiente, los componentes de los conjuntos  $Q$  y  $R$  dependen de la probabilidad de estar en los estados respectivos (proporciones de los vectores de aprendizaje en los grupos respectivos) y de las distribuciones de probabilidad para los grupos.

#### 5. Alternativas.

10 Muchos de los ejemplos precedentes implican la utilización de múltiples distribuciones/tablas para algunos valores de símbolo, y utilizan una única distribución/tabla para otros valores de símbolo. Si bien esta configuración reduce típicamente la utilización de memoria sin menoscabar significativamente el rendimiento de la codificación entrópica, las técnicas y herramientas descritas en la sección V son más generalmente aplicables a los modelos entrópicos organizados jerárquicamente. Un codificador o un decodificador pueden elegir selectivamente entre diferentes modelos entrópicos en una organización jerárquica que permita más adaptatividad para algunos valores de símbolo y menos adaptatividad para otros valores de símbolo.

15 Los modelos entrópicos organizados jerárquicamente pueden hacer referencia a múltiples modelos entrópicos por conmutación (por ejemplo, no conmutando solamente a un único modelo entrópico para valores de símbolo menos probables). Por ejemplo, un conjunto de tablas de códigos de Huffman, a cierto nivel comprende una tabla de códigos de Huffman o múltiples tablas de códigos de Huffman. El aprendizaje puede producirse en múltiples fases. En una primera fase de aprendizaje, los valores de símbolo son designados como perteneciendo a un conjunto  $Q$  o a un conjunto  $R$ , donde la distribución condicional para los valores de símbolo en el conjunto  $R$  es la misma para todos los estados. A continuación, en una fase de aprendizaje subsiguiente para los valores de símbolo del conjunto  $R$ , se levanta la restricción previa sobre la distribución condicional para los valores de símbolo en el conjunto  $R$ , y las distribuciones de probabilidad para los valores de símbolo del conjunto  $R$  son clasificadas en múltiples grupos/estados para diferentes modelos entrópicos.

25 Cada elemento de un conjunto de modelos entrópicos puede incluir múltiples puntos de conmutación a diferentes conjuntos de modelos entrópicos en otro nivel. Por ejemplo, para adaptación hacia delante, cada tabla de un primer conjunto de tablas de códigos de Huffman incluye dos códigos de escape (un primer código de escape a un segundo conjunto de una o varias tablas de códigos de Huffman, y un segundo código de escape a un tercer conjunto de una o varias tablas de códigos de Huffman). En relación con el aprendizaje, los valores de símbolo pueden ser designados como pertenecientes a un conjunto  $Q$  para un primer conjunto de modelos entrópicos, a un conjunto  $R$  para un segundo conjunto de modelos entrópicos, o a un conjunto  $S$  para un tercer conjunto de modelos entrópicos. La distribución condicional para valores de símbolo en el conjunto  $R$  (ignorando los valores de símbolo en  $Q$  y  $S$ ) es la misma para todos los estados, y la distribución condicional para valores de símbolo en el conjunto  $S$  (ignorando los valores de símbolo en  $Q$  y  $R$ ) es la misma para todos los estados.

35 Aparte de una amplitud adicional, los modelos entrópicos organizados jerárquicamente pueden incluir tres, cuatro o más niveles de modelos entrópicos. Por ejemplo, para adaptación hacia delante, cada tabla de un primer conjunto de tablas de códigos de Huffman incluye un código de escape a un segundo conjunto de tablas de códigos de Huffman, y cada tabla del segundo conjunto de códigos de Huffman incluye un código de escape a un tercer conjunto de tablas de códigos de Huffman. El aprendizaje puede producirse en múltiples fases. En una primera fase, los valores de símbolo son designados como perteneciendo a un conjunto  $Q$  para un primer conjunto de modelos entrópicos, o a un conjunto  $R$  para otros conjuntos de modelos entrópicos. La distribución condicional para valores de símbolo en el conjunto  $R$  (ignorando valores de símbolo en  $Q$ ) es igual para todos los estados. A continuación, en la fase de aprendizaje adicional para los valores de símbolo del conjunto  $R$ , se levanta esta restricción sobre la distribución condicional, y los valores de símbolo del conjunto  $R$  son designados como perteneciendo a un conjunto  $S$  para un segundo conjunto de modelos entrópicos, o a un conjunto  $T$  para cualquier otro conjunto de modelos entrópicos. En esta fase, la distribución condicional para valores de símbolo del conjunto  $T$  (ignorando valores de símbolo del conjunto  $S$ ) es la misma para todos los estados.

50 Aparte de codificación de longitud variable (por ejemplo, Huffman) y codificación y decodificación aritmética, otros tipos de codificación y decodificación entrópica pueden incorporar la utilización selectiva de modelos entrópicos. Por ejemplo, la codificación y decodificación variable a variable puede incorporar tablas VLC en una organización jerárquica.

#### C. Ejemplos de técnicas para codificación.

La figura 26 muestra una técnica 2600 para codificar símbolos con utilización selectiva de múltiples modelos entrópicos. Un codificador, tal como el codificador mostrado en la figura 2, 4 ó 6, lleva a cabo la técnica 2600.

En un codificador de audio de forma de onda, los símbolos son típicamente para coeficientes espectrales cuantificados. Los coeficientes espectrales cuantificados pueden ser preprocesados (por ejemplo, mediante predicción de coeficientes o reordenación de coeficientes). Cada uno de los símbolos puede representar un coeficiente espectral cuantificado. O bien, cada uno de los símbolos puede representar un grupo de coeficientes espectrales cuantificados. Para codificación Huffman vectorial, un símbolo representa, por ejemplo, un grupo de 4 coeficientes espectrales cuantificados. Para codificación de serie-nivel, un símbolo representa, por ejemplo, un par de serie-nivel.

Para una serie de símbolos, el codificador selecciona 2610 un modelo entrópico a partir de un primer conjunto de modelos entrópicos. Por ejemplo, el codificador selecciona una tabla de códigos Huffman a partir múltiples tablas de códigos Huffman disponibles, para codificación Huffman o codificación de serie-nivel. Alternativamente, el codificador selecciona un modelo entrópico utilizado en otro esquema de codificación entrópica. En algunas implementaciones, el codificador selecciona el modelo entrópico dependiendo de información contextual. En otras implementaciones, el codificador selecciona el modelo entrópico después de evaluar el rendimiento de la codificación utilizando los diversos modelos entrópicos. A continuación, se describe un ejemplo de un proceso de selección para tablas de códigos Huffman utilizando una estructura de enrejado. Alternativamente, el codificador utiliza otro mecanismo para seleccionar el modelo entrópico.

Volviendo la figura 26, el codificador señala opcionalmente 2620 información que indica el modelo entrópico seleccionado. Para adaptación hacia adelante, el codificador señala explícitamente información que indica el modelo entrópico seleccionado. A continuación se describe en detalle un mecanismo de adaptación hacia adelante para conmutación de tablas de códigos Huffman. Alternativamente, el codificador utiliza otro mecanismo de señalización. Para adaptación hacia atrás, la selección del modelo entrópico se deduce del contexto disponible en el descodificador.

A continuación, el codificador codifica entrópicamente 2360 la serie de símbolos, utilizando el modelo entrópico seleccionado. En cualesquiera puntos de conmutación en el modelo entrópico, el codificador puede conmutar a otro conjunto de uno o varios modelos entrópicos. Por ejemplo, el codificador utiliza un código de escape en una primera tabla de códigos de Huffman, para señalar una conmutación a una segunda tabla de códigos de Huffman, y a continuación codifica un símbolo utilizando la segunda tabla de códigos de Huffman.

A continuación, el codificador señala 2640 los símbolos codificados entrópicamente. Cuando se ha producido una conmutación, el codificador puede señalar asimismo información de conmutación, tal como códigos de escape u otra información de conmutación de modelo, para la selección dentro de un conjunto de modelos.

El codificador determina 2650 si continuar con la siguiente serie y, en caso afirmativo, selecciona 2610 el modelo entrópico para los símbolos de la serie siguiente. Por ejemplo, cuando se codifican coeficientes espectrales cuantificados utilizando tablas de códigos de Huffman en la implementación, se permite al codificador cambiar las tablas de códigos en límites de corteza. En otras palabras, los límites de corteza, que dividen el espectro de frecuencias, actúan como posibles posiciones de cambio para cambiar la tabla de códigos de Huffman seleccionada a partir de un primer conjunto de tablas de códigos. Si los coeficientes para el símbolo que está siendo codificado actualmente se extienden pasado un límite de corteza (por ejemplo, debido a que el símbolo representa un vector de coeficientes de un par de serie-nivel de coeficientes, que cruza el límite), entonces el final de los coeficientes para el símbolo actual resulta ser la posición de cambio válida. Alternativamente, el codificador cambia la selección del modelo entrópico desde el primer conjunto de modelos, en otras posiciones de cambio, y la serie de símbolos codificados de acuerdo con el modelo entrópico seleccionado tiene alguna otra duración.

Tal como se indicado anteriormente, en una implementación, el codificador selecciona una tabla de códigos de Huffman utilizando una estructura de enrejado para la evaluación de las diferentes tablas. El codificador codifica la totalidad de los símbolos entre dos posiciones válidas de cambio de tabla (que son límites de corteza), con todas las tablas posibles. El codificador realiza un seguimiento del número de bits utilizados por tabla para codificar los símbolos. El codificador constituye un enrejado para encontrar la mejor codificación posible, teniendo en cuenta los bits a señalar si se cambia una tabla.

Supóngase que  $b_{t,i}$  es el número mínimo de bits utilizados cuando se codifica hasta la posición  $t$  de cambio de tabla, siendo la tabla  $i$  la última tabla utilizada. El cómputo de bits  $r_{t,i}$  son los bits necesarios para codificar los símbolos entre la posición de cambio  $t$  y la posición de cambio  $t + 1$  utilizando la tabla  $i$ . El cómputo de bits  $s_{t,i,k}$  son los bits necesarios para codificar un cambio de tabla desde la tabla  $i$  a la tabla  $k$ , en la posición de cambio  $t$ . En otras palabras, la última tabla que se estaba utilizando en la posición de cambio  $t$  fue la tabla  $i$ , y a continuación se utiliza la tabla  $k$  para codificar hasta la posición de cambio  $t + 1$ . La tabla  $n_{t,i}$  es la tabla utilizada en la posición de cambio  $t - 1$  con objeto de obtener la codificación óptima en la cual la tabla actual en la posición de cambio  $t$  es la tabla  $i$ . Then:

$$n_{t+1,i} = \arg \min_k (b_{t,k} + r_{t,i} + s_{t,i,k})$$

$$b_{t+1,i} = \min_k (b_{t,k} + r_{t,i} + s_{t,i,k}) \quad (4).$$

El codificador determina la codificación óptima para la subtrama completa u otra parte de una secuencia, encontrando la  $i$  que minimiza  $b_{t_{\max},i}$ , donde  $t_{\max}$  es el valor máximo para  $t$ . El codificador encuentra las tablas óptimas rastreando el trayecto óptimo, mediante buscar el valor de  $n$ . Los bits necesarios para codificar un cambio de tabla son esencialmente  $\log_2(\text{número\_de\_tablas}) + \log_2(\text{número\_de\_corteza\_pasadas}) + 1$ . Cuando una tabla es modificada, el codificador señala un bit para indicar si ésta es la última tabla utilizada y, si no es la última tabla utilizada, el codificador señala  $\log_2(\text{número\_de\_corteza\_pasadas})$  para codificar a cuántas bandas de corteza aplica la tabla.

**D. Ejemplos de técnicas para descodificación.**

La figura 27 muestra una técnica 2700 para descodificar símbolos con utilización selectiva de múltiples modelos entrópicos. Un codificador, tal como el descodificador mostrado en la figura 3, 5 ó 7, lleva a cabo la técnica 2700.

En un descodificador de audio de forma de onda, los símbolos son típicamente para coeficientes espectrales cuantificados. Si los coeficientes espectrales cuantificados han sido preprocesados (por ejemplo, mediante predicción de coeficientes o reordenación de coeficientes) durante la descodificación, los coeficientes son postprocesados (por ejemplo, mediante predicción de coeficientes o reordenación de coeficientes) a continuación de la descodificación entrópica. Cada uno de los símbolos puede representar un coeficiente espectral cuantificado. O bien, cada uno de los símbolos puede representar un grupo de coeficientes espectrales cuantificados. Para descodificación Huffman vectorial, un símbolo representa, por ejemplo, un grupo de 4 coeficientes espectrales cuantificados. Para descodificación de serie-nivel, un símbolo representa, por ejemplo, un par de serie-nivel.

Para una serie de símbolos, opcionalmente el descodificador analiza sintácticamente 2710 información que indica el modelo entrópico seleccionado. Para adaptación hacia delante, por ejemplo, el descodificador analiza sintácticamente información que indica el modelo entrópico seleccionado, utilizando un mecanismo que refleja la señalización del lado del codificador.

El descodificador selecciona 2720 un modelo entrópico a partir de un primer conjunto de modelos entrópicos. Por ejemplo, el descodificador selecciona una tabla de códigos Huffman entre múltiples tablas disponibles de códigos Huffman, para descodificación vectorial Huffman o descodificación de serie-nivel. Alternativamente, el descodificador selecciona un modelo entrópico utilizado en otro esquema de descodificación entrópica. En algunas implementaciones, el descodificador selecciona el modelo entrópico dependiendo de información contextual para adaptación hacia atrás. En otras implementaciones, el descodificador selecciona el modelo entrópico en base a información señalizada mediante un codificador y analizada sintácticamente 2710 a partir del flujo de bits.

A continuación, el descodificador descodifica entrópicamente 2730 la serie de símbolos, utilizando el modelo entrópico seleccionado. En cualesquiera puntos de conmutación en el modelo entrópico, el descodificador puede conmutar a otro conjunto de uno o varios modelos entrópicos. Por ejemplo, el descodificador recibe un código de escape para una primera tabla de códigos de Huffman, que indica una conmutación a una segunda tabla de códigos de Huffman, y a continuación codifica el símbolo utilizando la segunda tabla de códigos de Huffman.

A continuación, el codificador entrega 2740 información para los símbolos descodificados entrópicamente, por ejemplo, los coeficientes espectrales cuantificados listos para el procesamiento posterior.

El descodificador determina 2750 si continuar con la siguiente serie y, en caso afirmativo, selecciona 2710 el modelo entrópico para los símbolos de la serie siguiente. Por ejemplo, cuando se descodifican coeficientes espectrales cuantificados utilizando tablas de códigos de Huffman en la implementación, se permite al descodificador cambiar las tablas de códigos en límites de corteza. Si los coeficientes para el símbolo que está siendo descodificado actualmente se extienden pasado un límite de corteza (por ejemplo, debido a que el símbolo representa un vector de coeficientes de un par de serie-nivel de coeficientes que cruza el límite), entonces la finalización de los coeficientes para el símbolo actual resulta la posición de cambio válida. Alternativamente, el descodificador cambia la selección del modelo entrópico a partir del primer conjunto de modelos en otras posiciones de cambio, y la serie de símbolos descodificados de acuerdo con el modelo entrópico seleccionado tiene alguna otra duración.

**E. Resultados.**

La codificación utilizando una distribución aproximada para valores de símbolo menos probables, permite ahorros de memoria necesarios para las distribuciones o tablas de código, en el codificador y el descodificador. En términos del marco analítico de la sección V.B.1, el codificador y el descodificador almacenan las distribuciones y/o tablas de

código para  $P_{S(j),X(i)}$ . Es decir, el codificador y el decodificador almacenan una distribución y/o tabla por estado  $S(j)$  para valores de símbolo  $X(i)$  en el conjunto  $Q$ . Para valores de símbolo  $X(i)$  en el conjunto  $R$ , el codificador y el decodificador almacenan la distribución y/o la tabla para una sola distribución  $P'_{S(j),X(i),R}$ .

5 Supóngase que una tabla ocupa  $B$  octetos de memoria para cada estado, y que existen 16 estados. Entonces, en el típico caso de tablas completas, el codificador y el decodificador necesitarían  $16 * B$  octetos de memoria para las 16 tablas. Sin embargo, si solamente el 10% de los valores de símbolos se designan como más probables (en el conjunto  $Q$ ), entonces una aproximación simple de la memoria necesaria es  $(16 * B * 0,1) + (B * 0,9) = 2,5 * B$ . Por lo tanto, la memoria necesaria se ha reducido en más de 6 veces, con solamente una ligera reducción en las ganancias de codificación entrópica, en comparación con el caso de tablas completas.

10 A la vista de las muchas realizaciones posibles a las que pueden ser aplicados los principios de la invención dada a conocer, debe admitirse que las realizaciones ilustradas son solamente ejemplos preferidos de la invención, y no deben tomarse como limitativos del alcance de la invención. Por el contrario, el alcance de la invención está definido por las siguientes reivindicaciones. Por lo tanto, reivindicamos como nuestra invención todo lo que queda dentro del alcance de estas reivindicaciones.

15

**REIVINDICACIONES**

1. Método para ser ejecutado por un codificador o un decodificador, que comprende:

5 para una serie de símbolos, seleccionar un modelo entrópico a partir de un primer conjunto de modelos que incluye múltiples modelos entrópicos, cada uno de los múltiples modelos entrópicos del primer conjunto de modelos incluyendo un punto de conmutación de modelo para conmutar a un segundo conjunto de modelos que incluye uno o varios modelos entrópicos, en el que la serie de símbolos son para coeficientes espectrales cuantificados para datos de audio;

y en el que la selección de un modelo entrópico se basa en la evaluación del rendimiento de la codificación utilizando los múltiples modelos entrópicos;

10 procesar la serie de símbolos utilizando el modelo entrópico seleccionado; y

entregar los resultados del proceso;

15 en el que los múltiples modelos entrópicos del primer conjunto de modelos, y dichos uno o varios modelos entrópicos del segundo conjunto de modelos reflejan distribuciones de probabilidad para codificación y/o descodificación aritmética, y en el que los múltiples modelos entrópicos del primer conjunto de modelos reflejan las distribuciones de probabilidad de los símbolos más probables, y los múltiples modelos entrópicos del segundo conjunto de modelos reflejan las distribuciones de probabilidad de los símbolos menos probables;

y en el que es seguido un punto de conmutación si la distribución de probabilidad seleccionada no comprende el símbolo respectivo de la serie de símbolos.

20 2. El método acorde con la reivindicación 1, en el que el procesamiento incluye codificación entrópica si el método es ejecutado por un codificador.

3. El método acorde con la reivindicación 1, en el que el procesamiento incluye descodificación entrópica si el método es ejecutado por un decodificador.

4. El método acorde con la reivindicación 1, en el que el punto de conmutación de modelos es una probabilidad de conmutación de modelos en las múltiples distribuciones de probabilidad del primer conjunto de modelos.

25 5. El método acorde con la reivindicación 1, en el que los múltiples modelos entrópicos del primer conjunto de modelos están realizados, respectivamente, en las múltiples tablas VLC de un primer conjunto de tablas, en el que dichos uno o varios modelos entrópicos del segundo conjunto de modelos están realizados respectivamente en una o varias tablas VLC de un segundo conjunto de modelos, en el que el punto de conmutación de modelos es un código de escape, y en el que cada una de las múltiples tablas VLC del primer conjunto de tablas incluye el código de escape para conmutar al segundo conjunto de tablas.

35 6. El método acorde con la reivindicación 5, en el que las múltiples tablas VLC del primer conjunto de tablas y dichas una o varias tablas VLC del segundo conjunto de tablas son tablas de códigos de Huffman, y en el que el segundo conjunto de tablas incluye una sola tabla de códigos de Huffman, de tal modo que dicha única tabla de códigos de Huffman representa una rama común en los árboles que representan las respectivas múltiples tablas de códigos de Huffman del primer conjunto de tablas.

7. El método acorde con la reivindicación 5, en el que las múltiples tablas VLC del primer conjunto de tablas están adaptadas para un primer conjunto de valores de símbolo que incluye valores de símbolo más probables, y en el que dichas una o varias tablas VLC del segundo conjunto de tablas están adaptadas para un segundo conjunto de valores de símbolo que incluye valores de símbolo menos probables.

40 8. El método acorde con la reivindicación 7, en el que el segundo conjunto de tablas incluye una sola tabla VLC, y en el que el proceso es para codificación o descodificación de longitud variable en dos etapas, de aquellos de la serie de símbolos que tienen valores de símbolo menos probables.

45 9. El método acorde con la reivindicación 1, que comprende además generar los múltiples modelos entrópicos del primer conjunto de modelos y dichos uno o varios modelos entrópicos del segundo conjunto de modelos, en el que la generación incluye:

agrupar distribuciones de probabilidad de acuerdo con la primera métrica de costos, obteniendo como resultado una serie de grupos preliminares; y

refinar la serie de grupos preliminares de acuerdo con una segunda métrica de costos diferente a la primera métrica de costos, obteniendo como resultado una serie de grupos finales.

5 10. El método acorde con la reivindicación 1, en el que en el segundo conjunto de modelos incluye un solo modelo entrópico, comprendiendo además el método la generación de los múltiples modelos entrópicos del primer conjunto de modelos y del único modelo entrópico del segundo conjunto de modelos, en el que la generación incluye, para el único modelo entrópico del segundo conjunto de modelos, restringir los valores de símbolo menos probables a tener una distribución condicional común a través de las distribuciones de probabilidad.

10 11. El método acorde con la reivindicación 1, en el que cada uno de dichos uno o varios modelos entrópicos del segundo conjunto de modelos incluye un segundo punto de conmutación de modelos, para conmutar a un tercer conjunto de modelos que incluye uno o varios modelos entrópicos.

12. El método acorde con la reivindicación 1, en el que, para por lo menos parte de los múltiples modelos entrópicos del primer conjunto de modelos, el punto de conmutación de modelos tiene un valor diferente entre modelo y modelo.

15 13. El método acorde con la reivindicación 1, en el que cada uno de los múltiples modelos entrópicos del primer conjunto de modelos incluye además un segundo punto de conmutación de modelos para conmutar a un tercer conjunto de modelos que incluye uno o varios modelos entrópicos.

14. El método acorde con la reivindicación 1, en el que la selección forma parte de una conmutación adaptativa hacia adelante.

20 15. El método acorde con la reivindicación 1, en el que la selección forma parte de una conmutación adaptativa hacia atrás.

16. Un sistema que comprende un codificador y descodificador, comprendiendo además:

medios para obtener distribuciones de probabilidad para una serie de valores de símbolo, en el que los valores de símbolo son para coeficientes espectrales cuantificados para datos de audio; y

25 medios para generar modelos entrópicos, que incluyen restringir la serie de valores de símbolo menos probables, a tener una distribución condicional común a través de las distribuciones de probabilidad, sin restringir por ello la serie de valores de símbolo más probables,

30 medios para seleccionar un modelo entrópico a partir de un primer conjunto de modelos que incluye múltiples modelos entrópicos, cada uno de los múltiples modelos entrópicos del primer conjunto de modelos incluyendo un punto de conmutación de modelos para conmutar a un segundo conjunto de modelos que incluye uno o varios modelos entrópicos, en el que la selección de un modelo entrópico está basada en la evaluación del rendimiento de la codificación utilizando los múltiples modelos entrópicos;

medios para procesar dicha serie de símbolos utilizando el modelo entrópico seleccionado; y

medios para entregar los resultados del proceso;

35 en el que los múltiples modelos entrópicos del primer conjunto de modelos y dichos uno o varios modelos entrópicos del segundo conjunto de modelos reflejan distribuciones de probabilidad para codificación y/o descodificación aritmética; y

en el que los múltiples modelos entrópicos del primer conjunto de modelos reflejan las distribuciones de probabilidad de los símbolos más probables, y los múltiples modelos entrópicos del segundo conjunto de modelos reflejan las distribuciones de probabilidad de los símbolos menos probables; y

40 en el que se sigue un punto de conmutación si la distribución de probabilidad seleccionada no comprende el símbolo respectivo de la serie de valores de símbolo.

17. El sistema acorde con la reivindicación 16, en el que uno o varios módulos generan modelos entrópicos mediante:

45 agrupar distribuciones de probabilidad de acuerdo con la primera métrica de costos, obteniendo como resultado una serie de grupos preliminares;

refinar la serie de grupos preliminares de acuerdo con una segunda métrica de costos diferente a la primera métrica de costos, obteniendo como resultado una serie de grupos finales; y

seleccionar los modelos entrópicos en base, por lo menos en parte, a la serie de grupos finales.

18. El sistema acorde con la reivindicación 17, en el que la segunda métrica de costos es la entropía relativa.

- 5 19. El sistema acorde con la reivindicación 17, en el que los modelos entrópicos son realizados respectivamente en múltiples tablas VLC de un primer conjunto de tablas y en una tabla VLC única de un segundo conjunto de tablas, en el que las múltiples tablas VLC están adaptadas para la serie de valores de símbolo más probables, y en el que la tabla VLC única está adaptada para la serie de valores de símbolo menos probables.

Figura 1

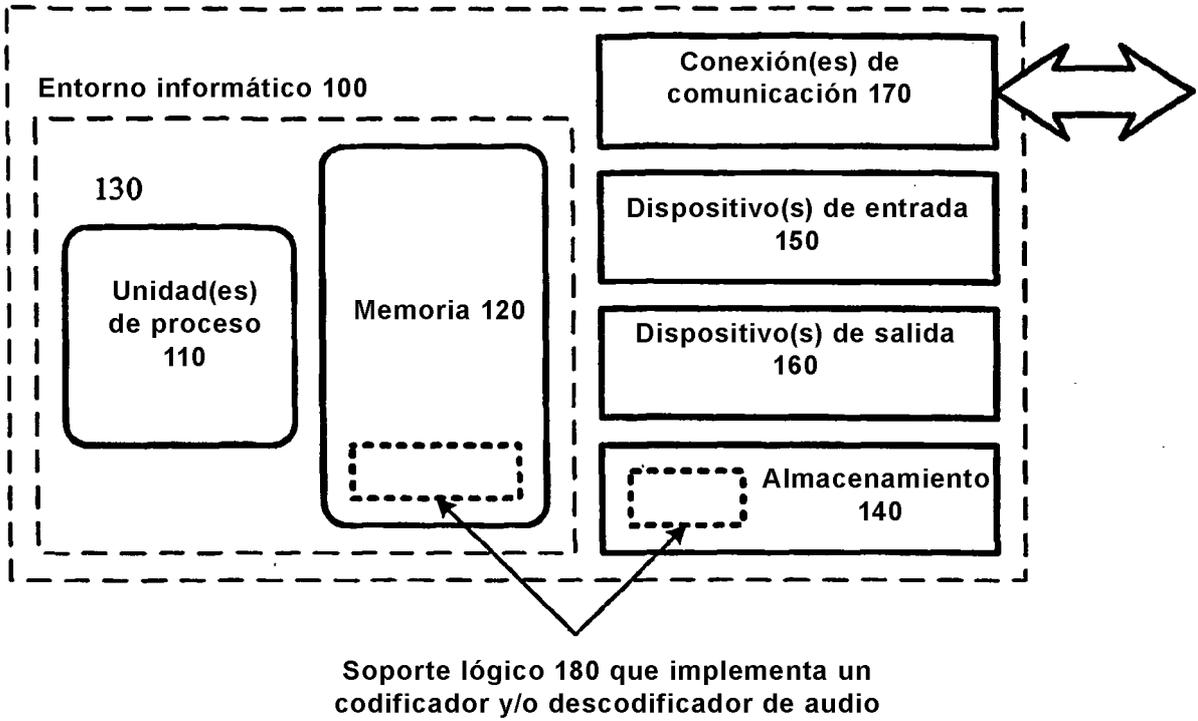


Figura 2

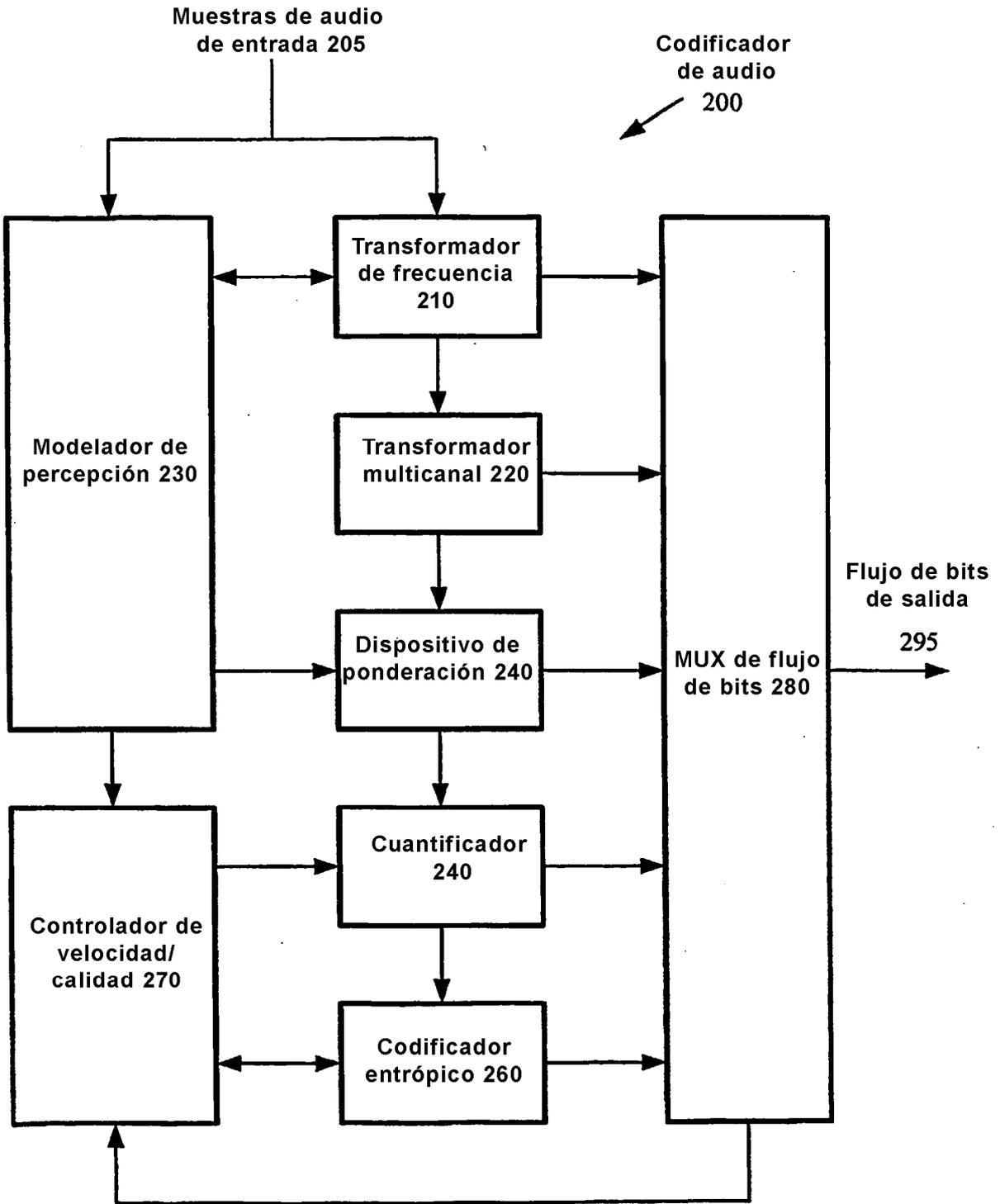


Figura 3

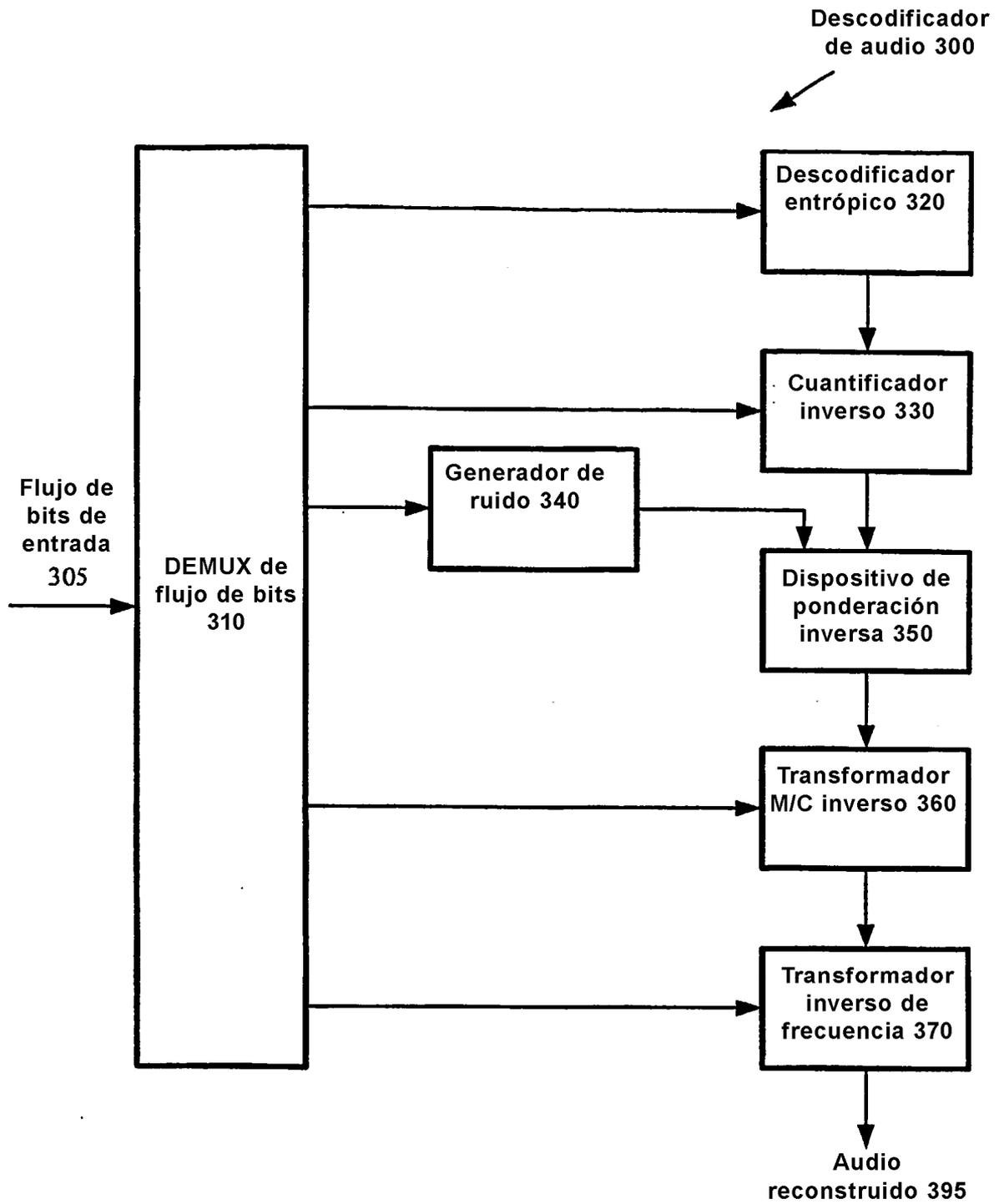


Figura 4

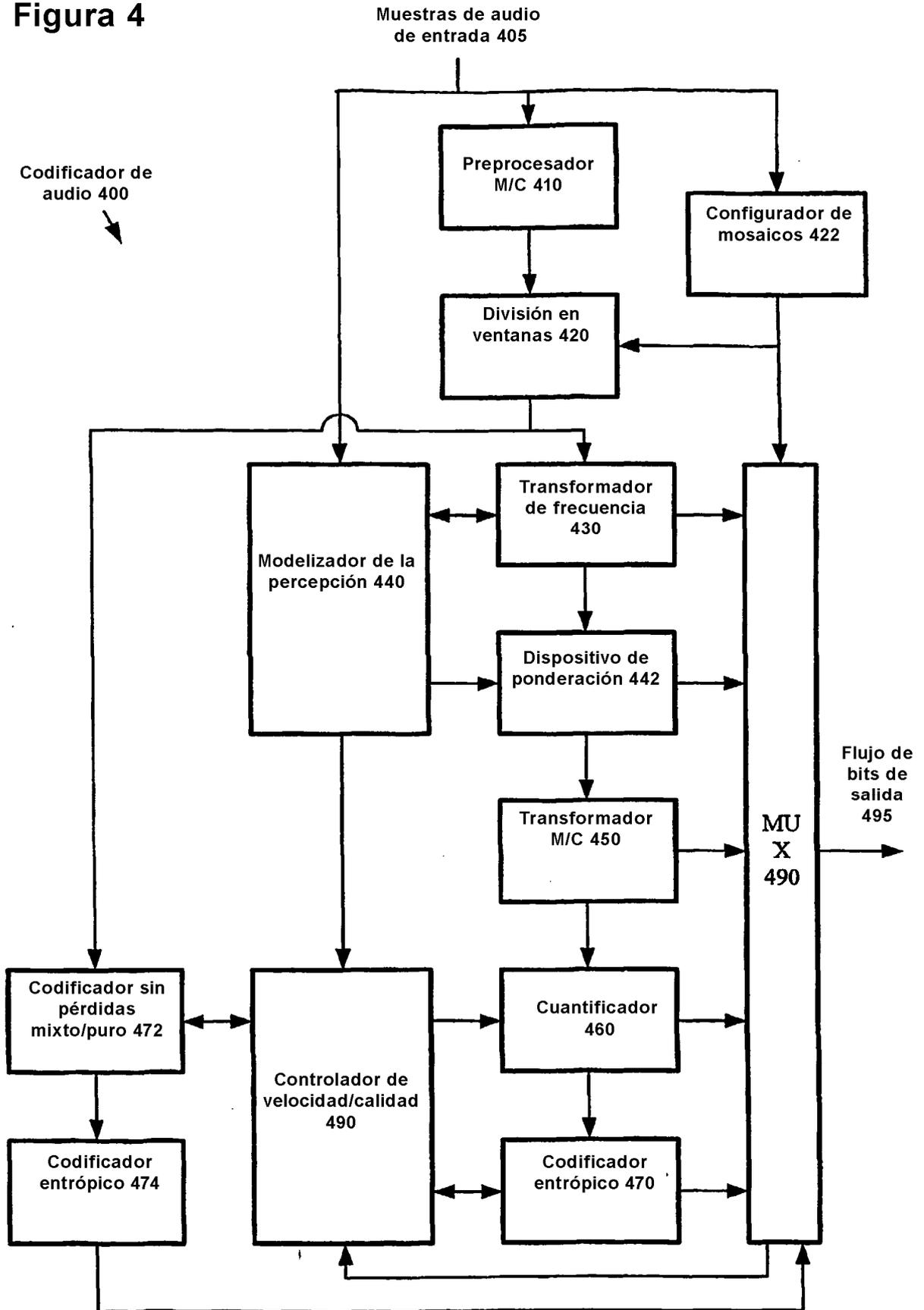
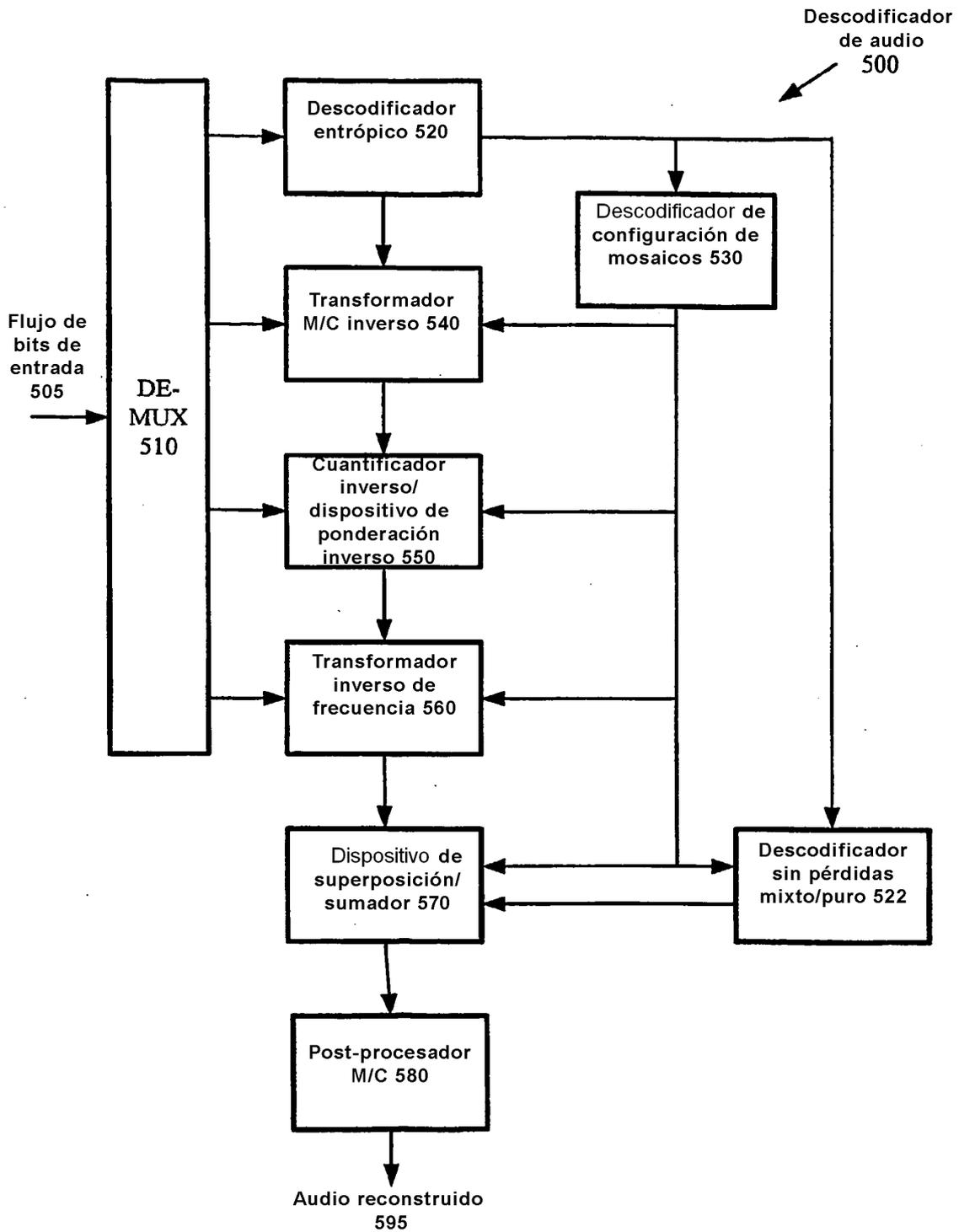
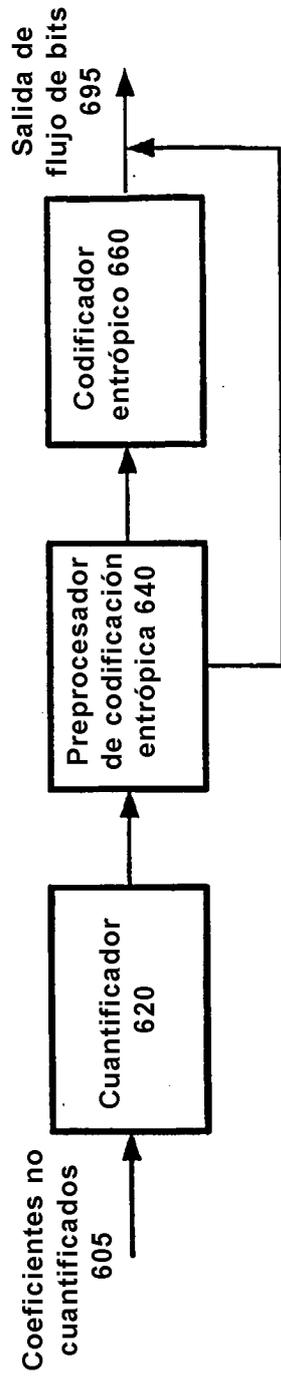


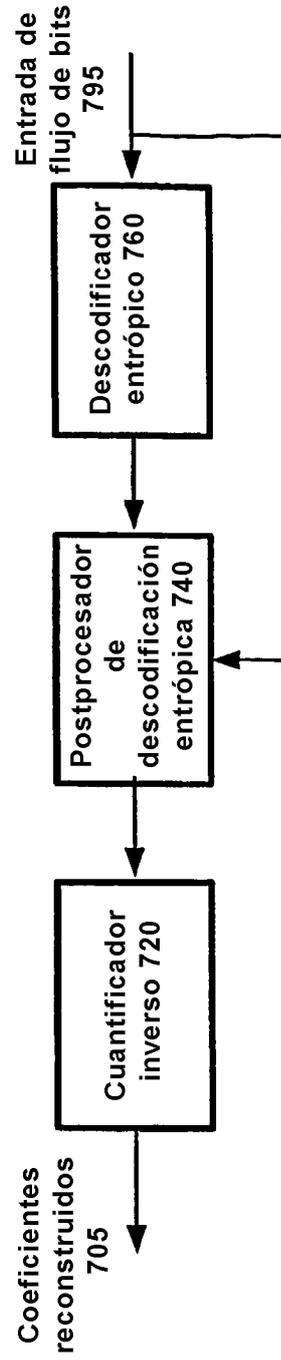
Figura 5



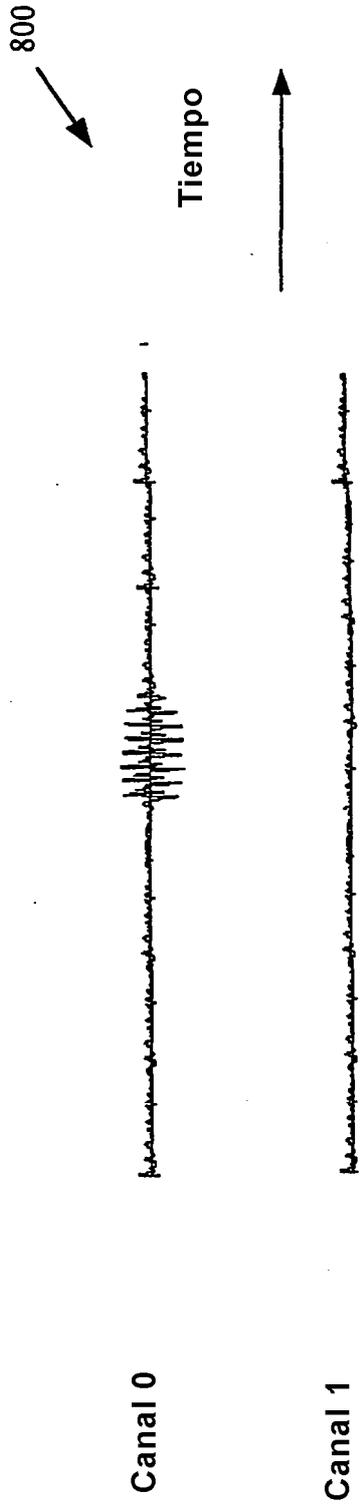
**Figura 6**



**Figura 7**



**Figura 8a**



**Figura 8b**

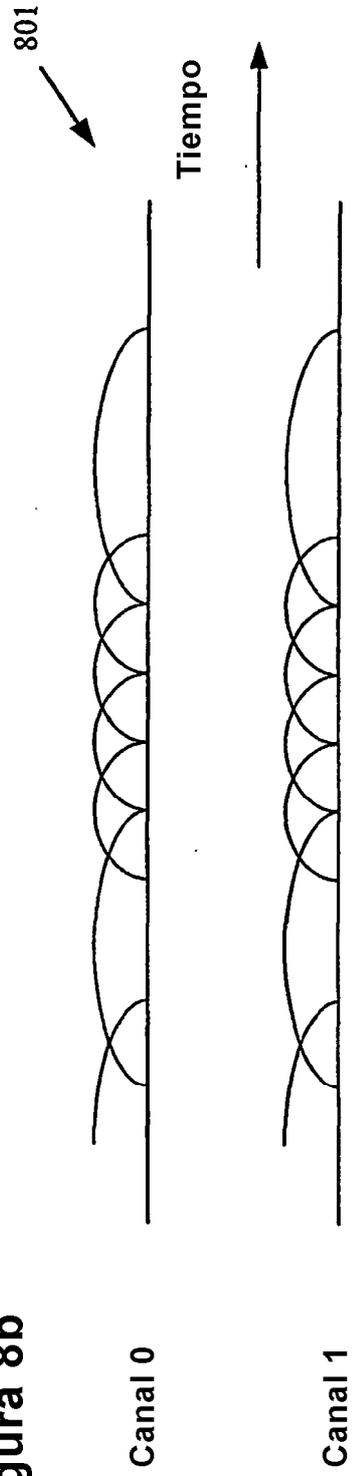


Figura 9

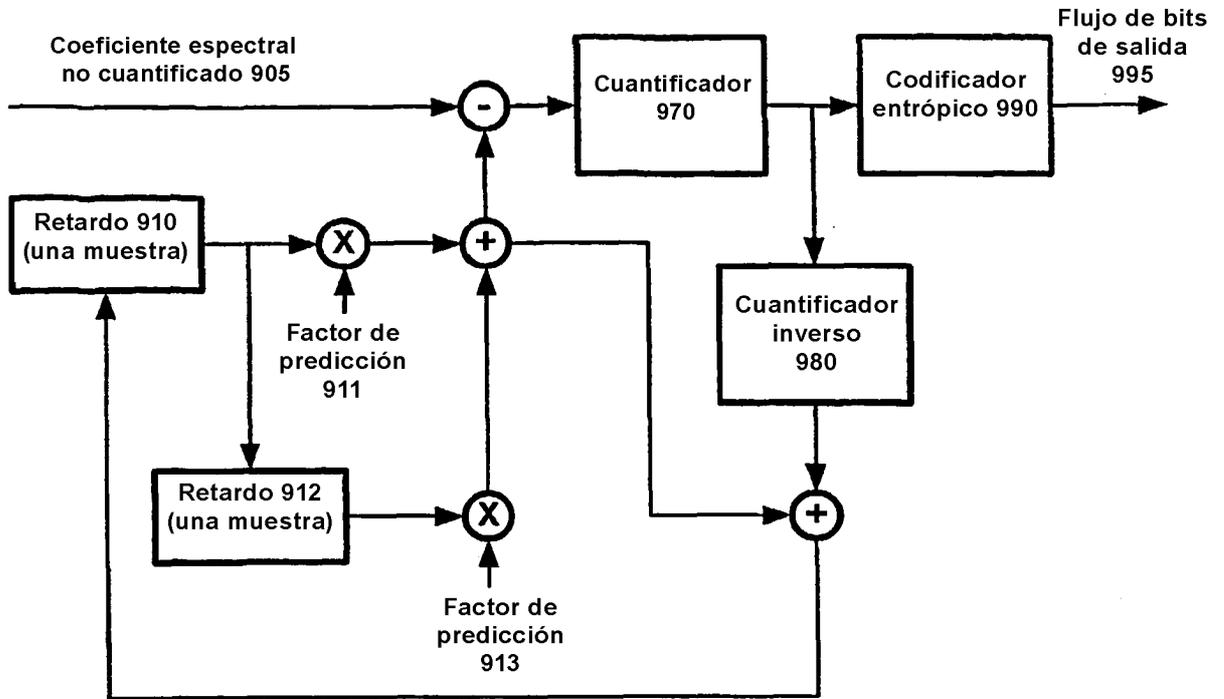


Figura 10

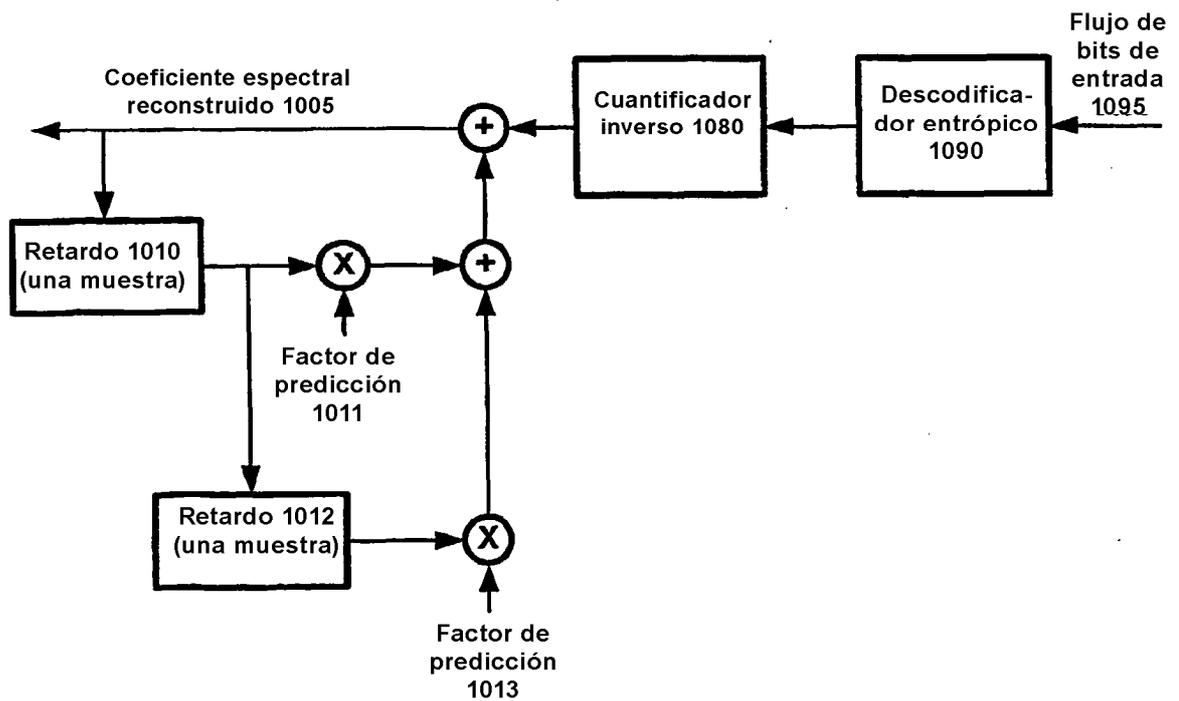


Figura 11

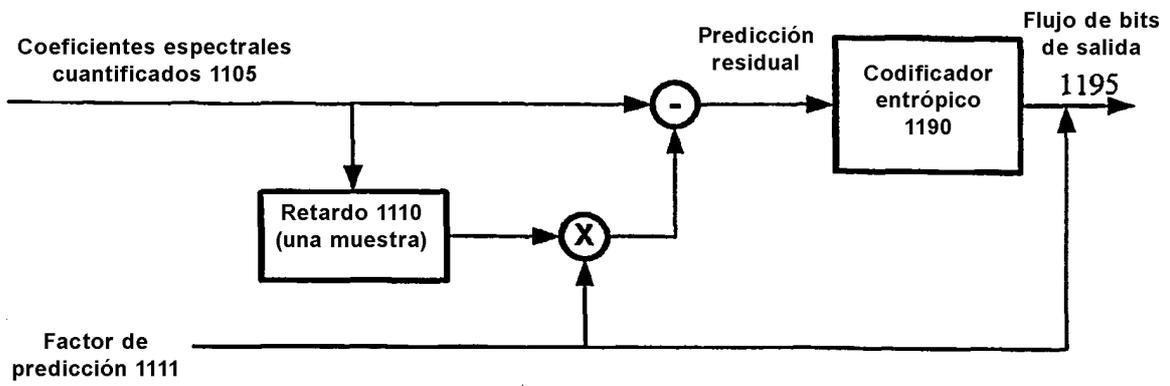


Figura 12

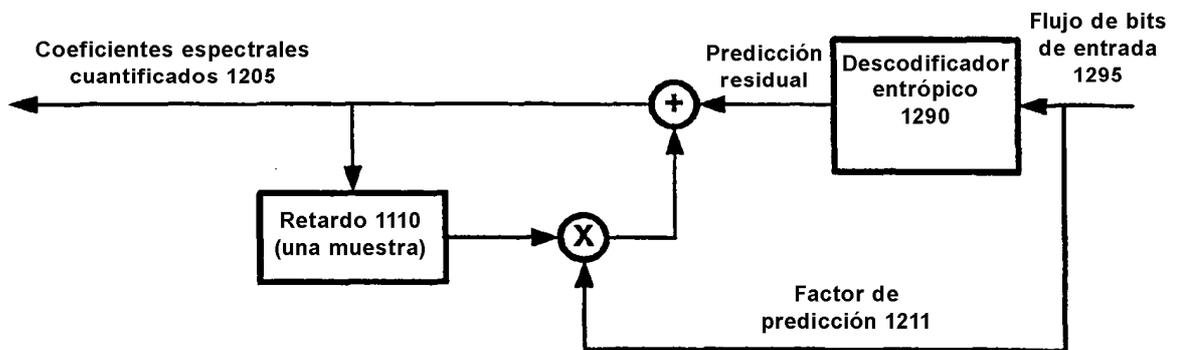


Figura 13

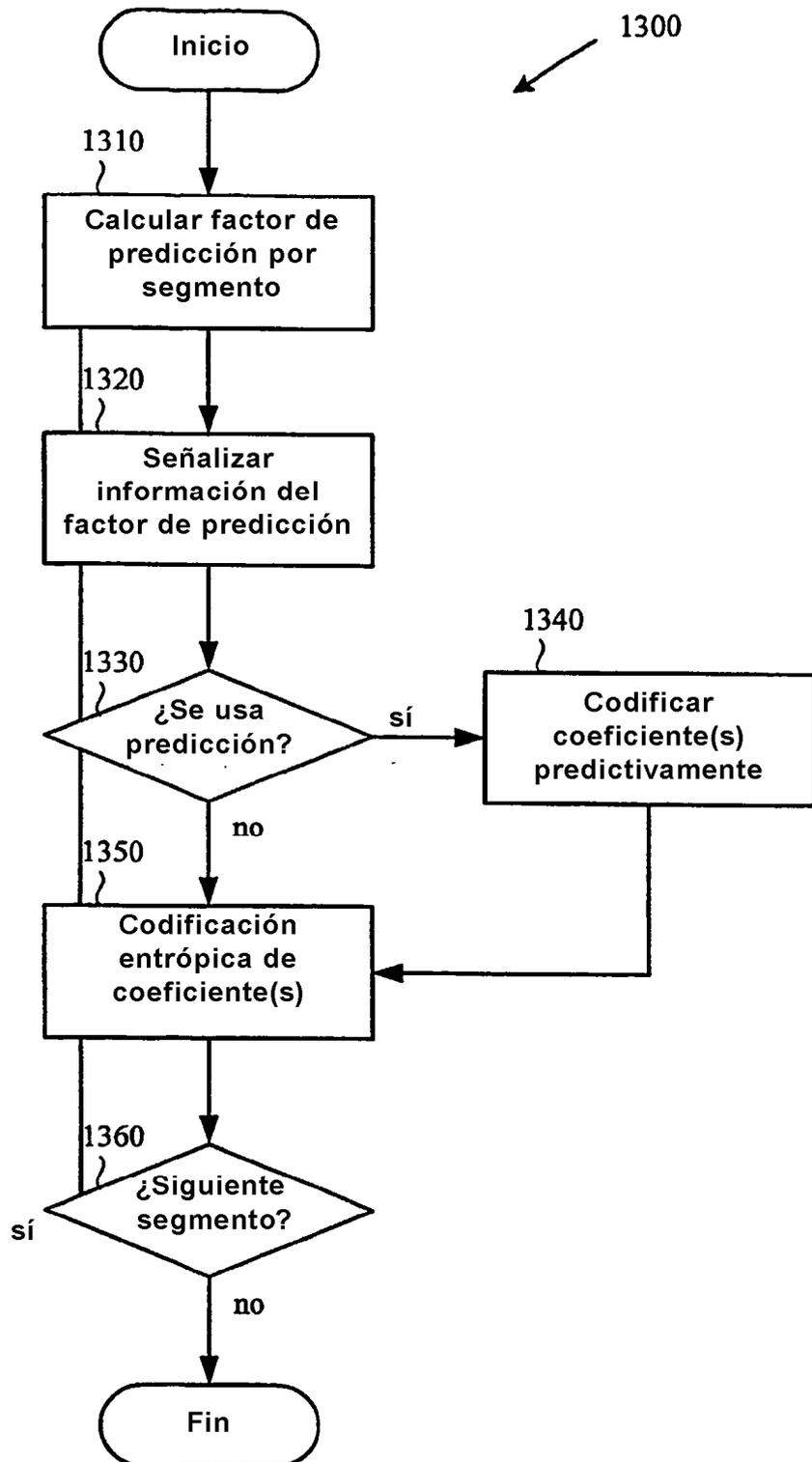


Figura 14

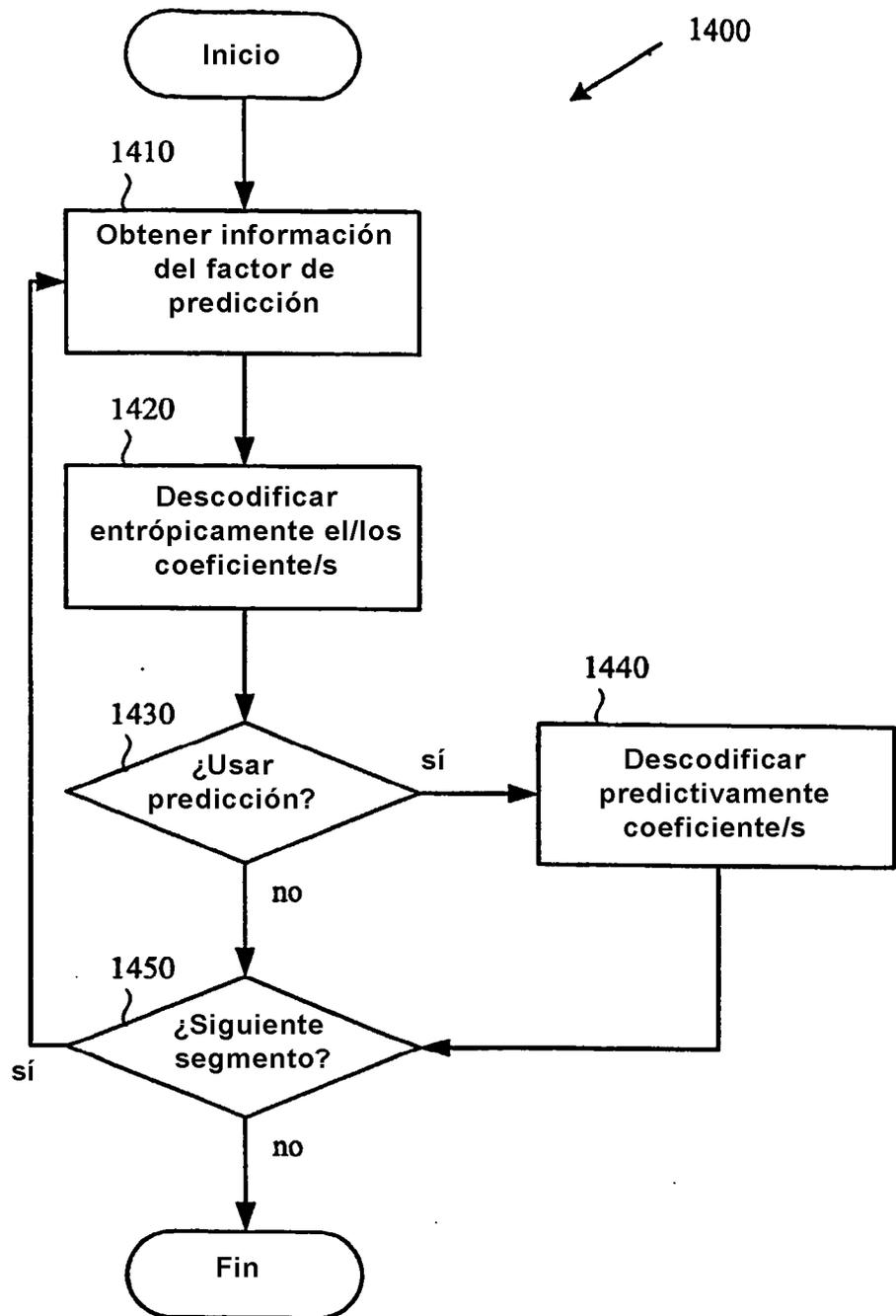


Figura 15a

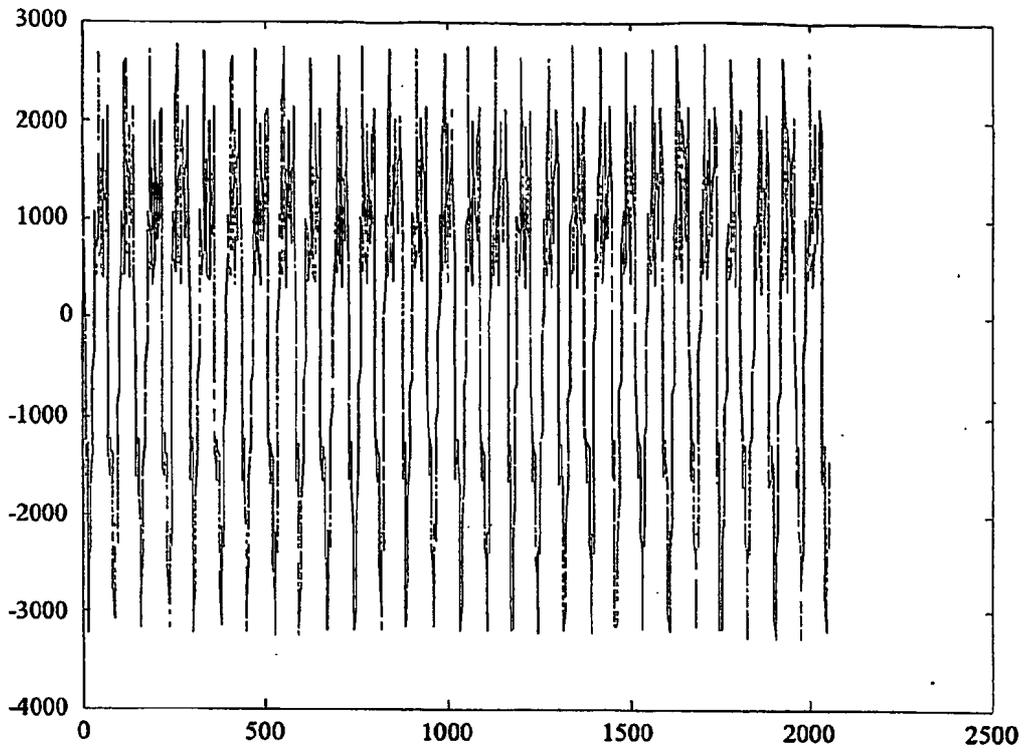
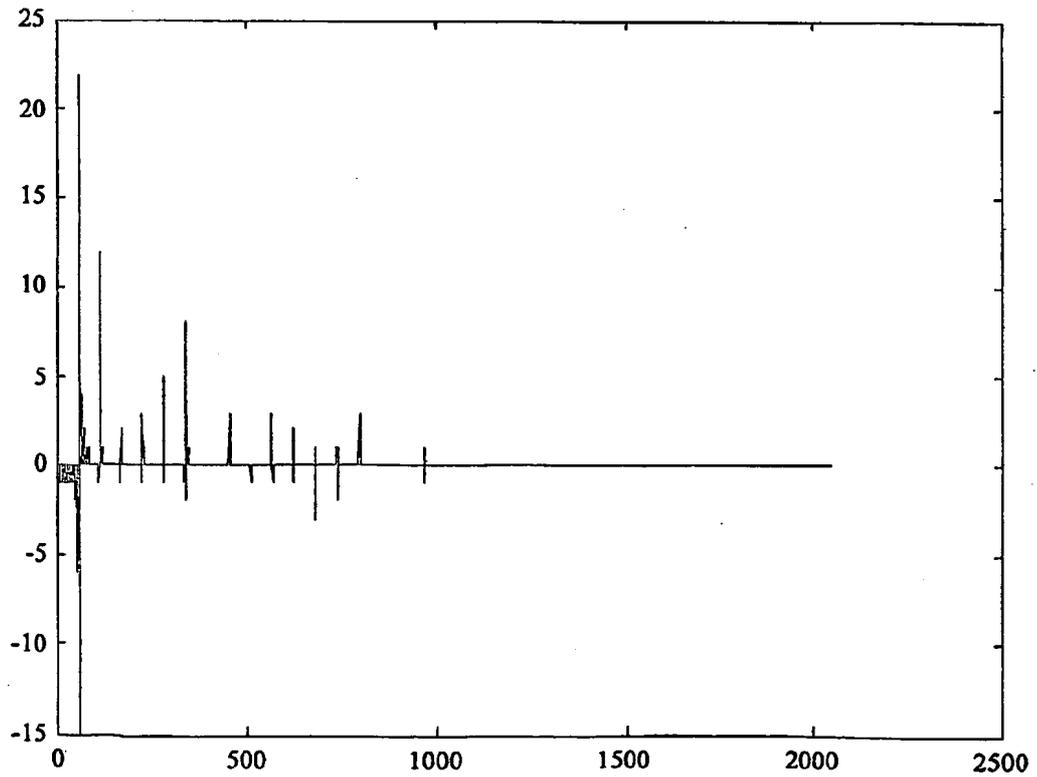
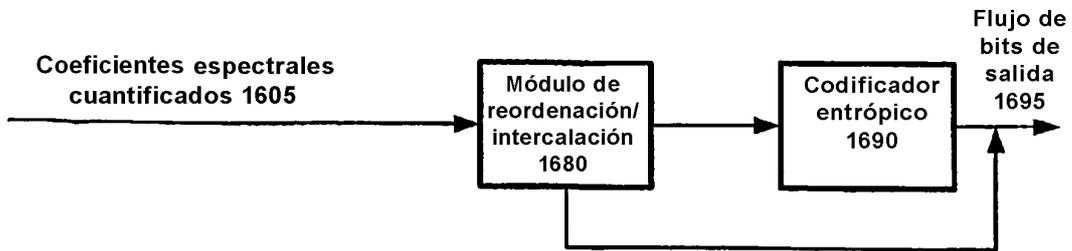


Figura 15b



**Figura 16**



**Figura 17**

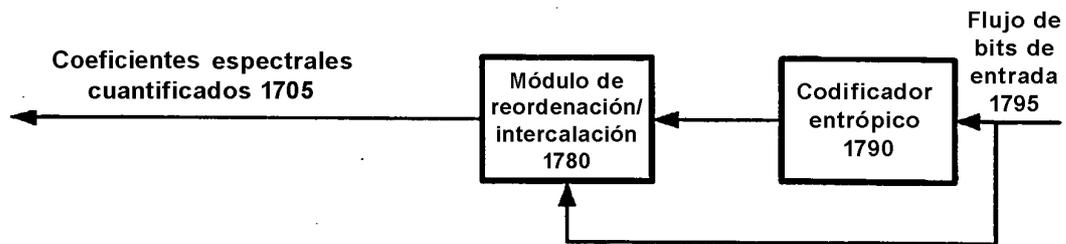


Figura 18a

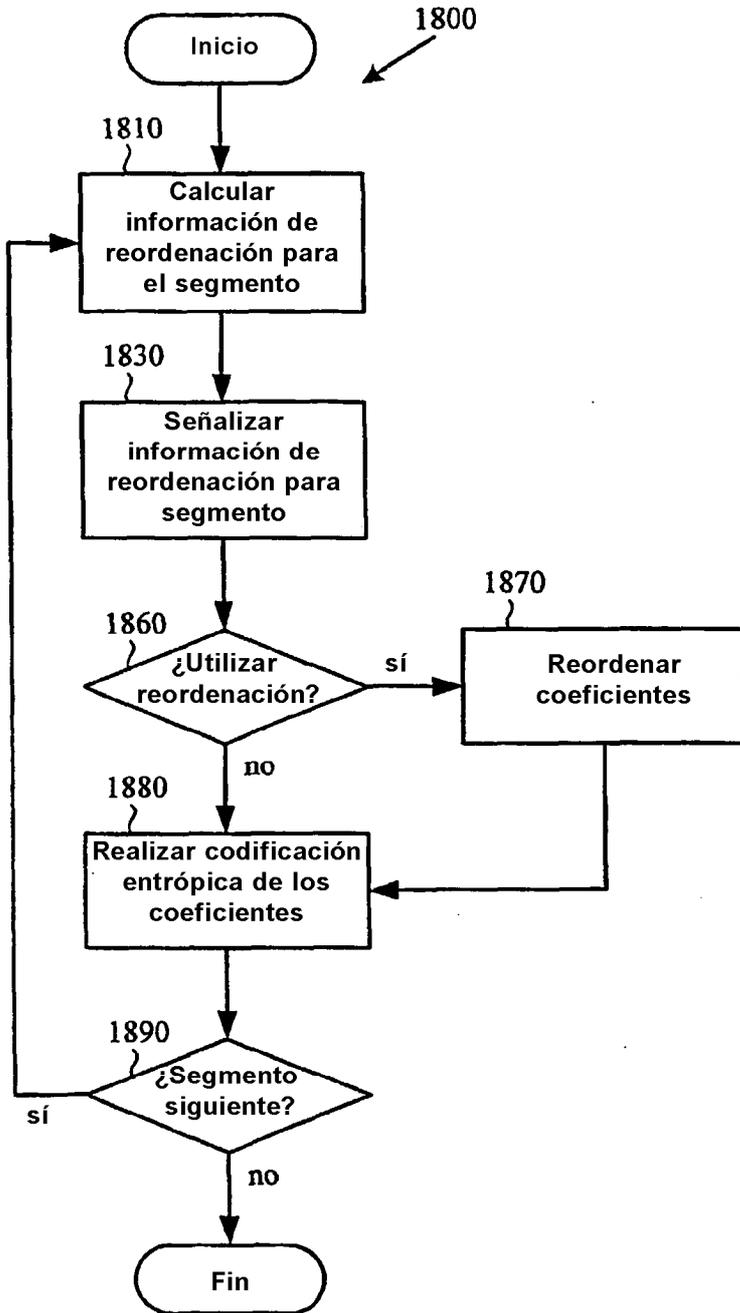


Figura 18b

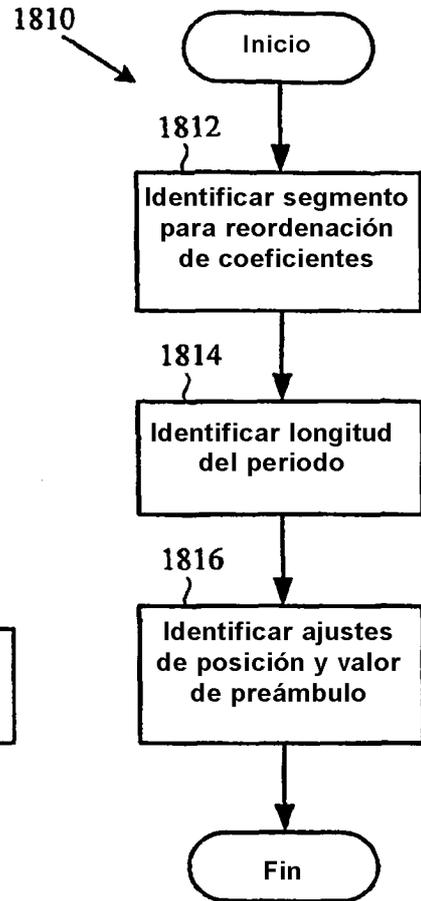


Figura 18c

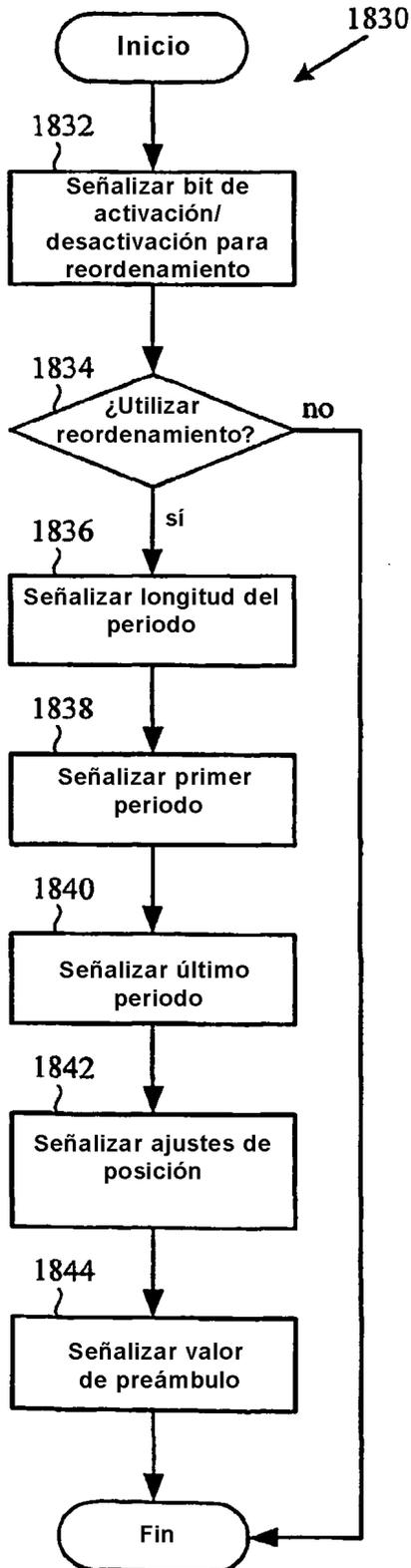


Figura 19b

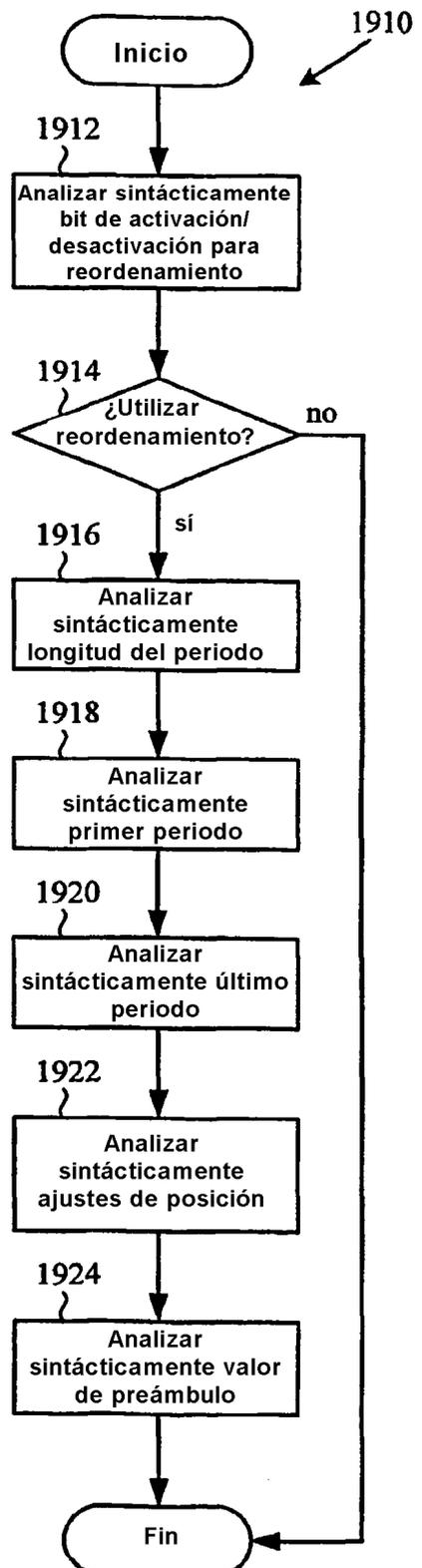


Figura 19a

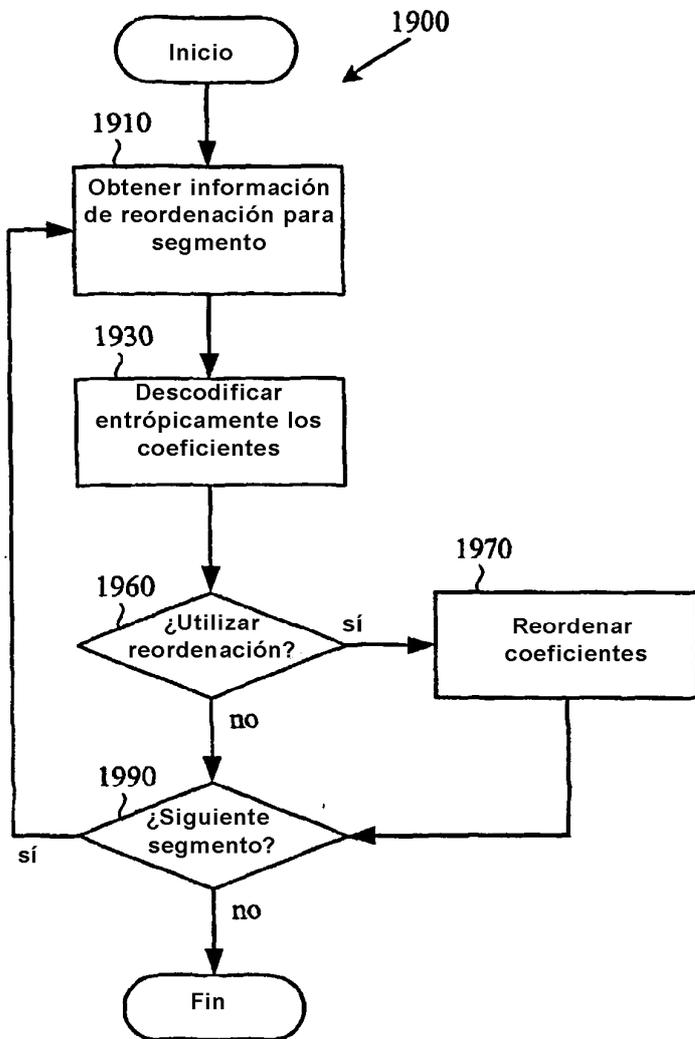


Figura 19c

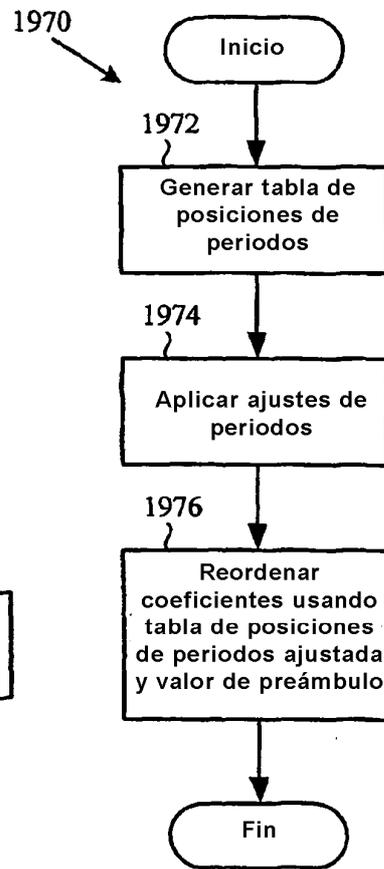


Figura 20

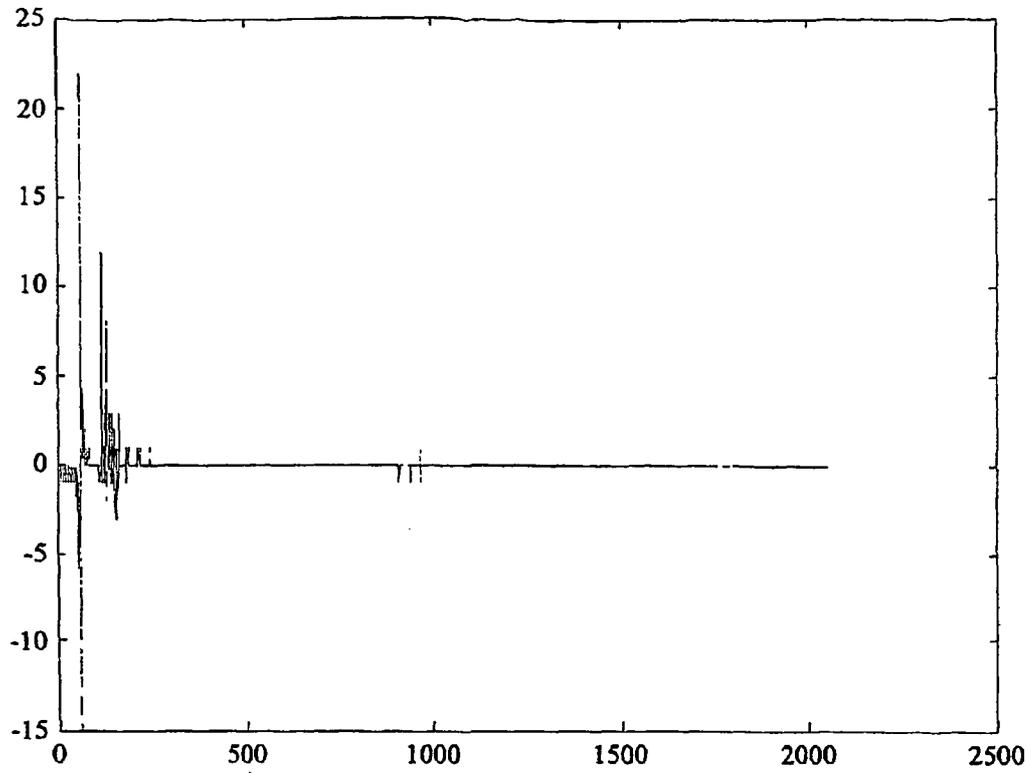
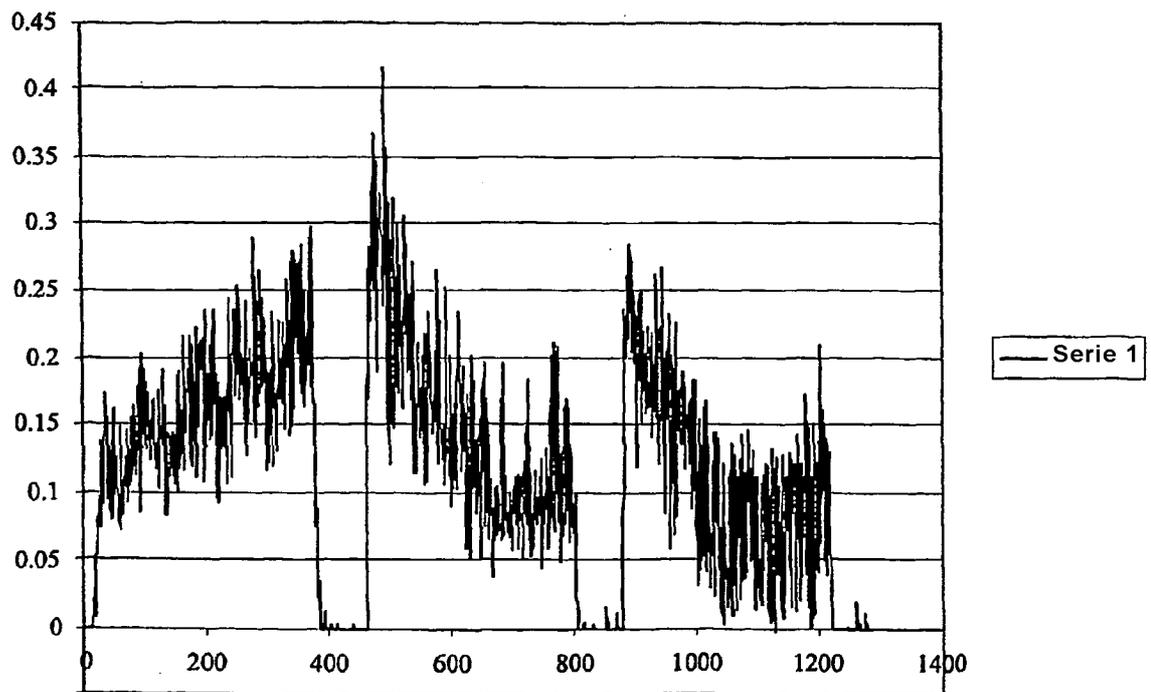
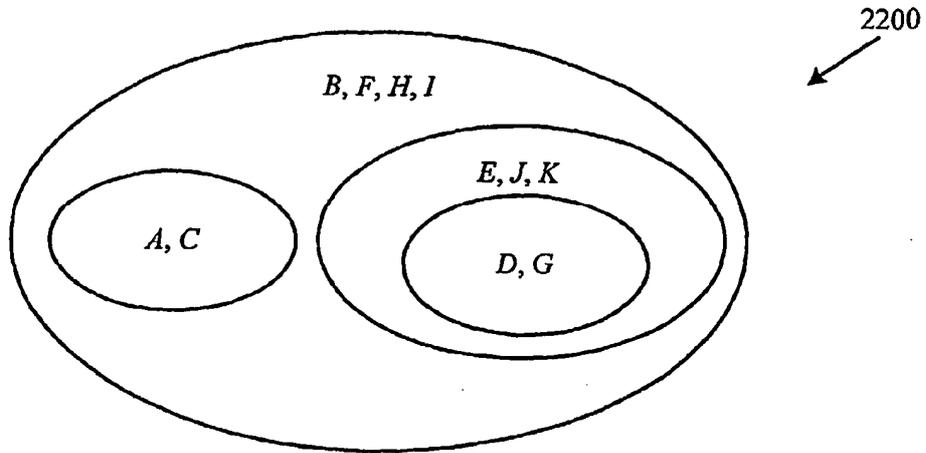


Figura 21



**Figura 22**



Primer conjunto de modelos: 8 modelos entrópicos para {B, F, H, I, conmutador\_1, conmutador\_2}

Segundo conjunto de modelos: 3 modelos entrópicos para {A, C} después de conmutador\_1

Tercer conjunto de modelos: 4 modelos entrópicos para {E, J, K, conmutador\_3} después de conmutador\_2

Cuarto conjunto de modelos: 1 modelo entrópico para {D, G} después de conmutador\_3

**Figura 23**

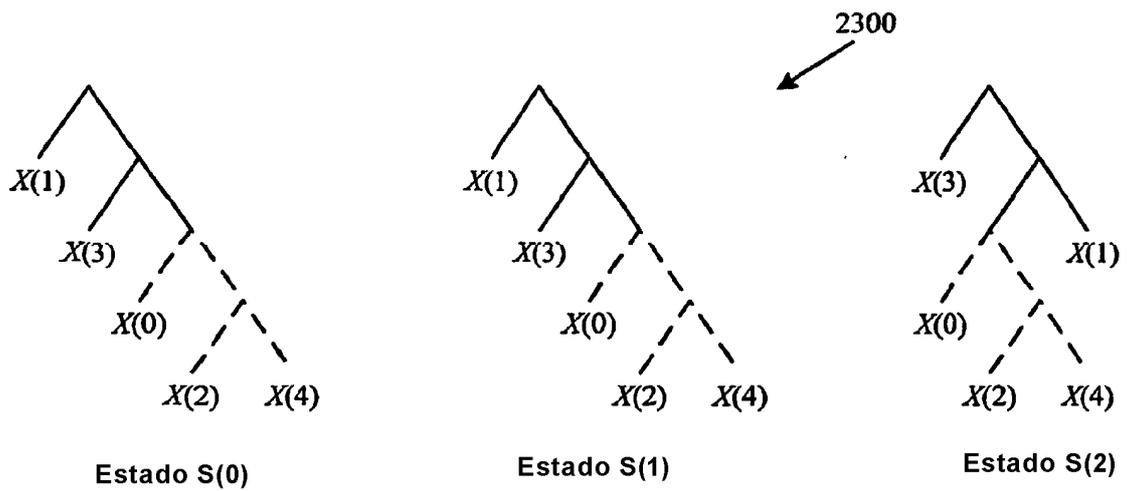


Figura 24

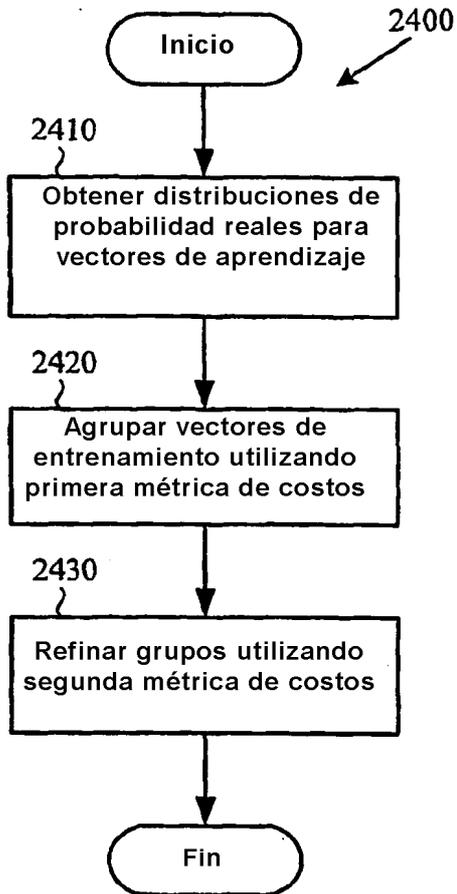


Figura 25

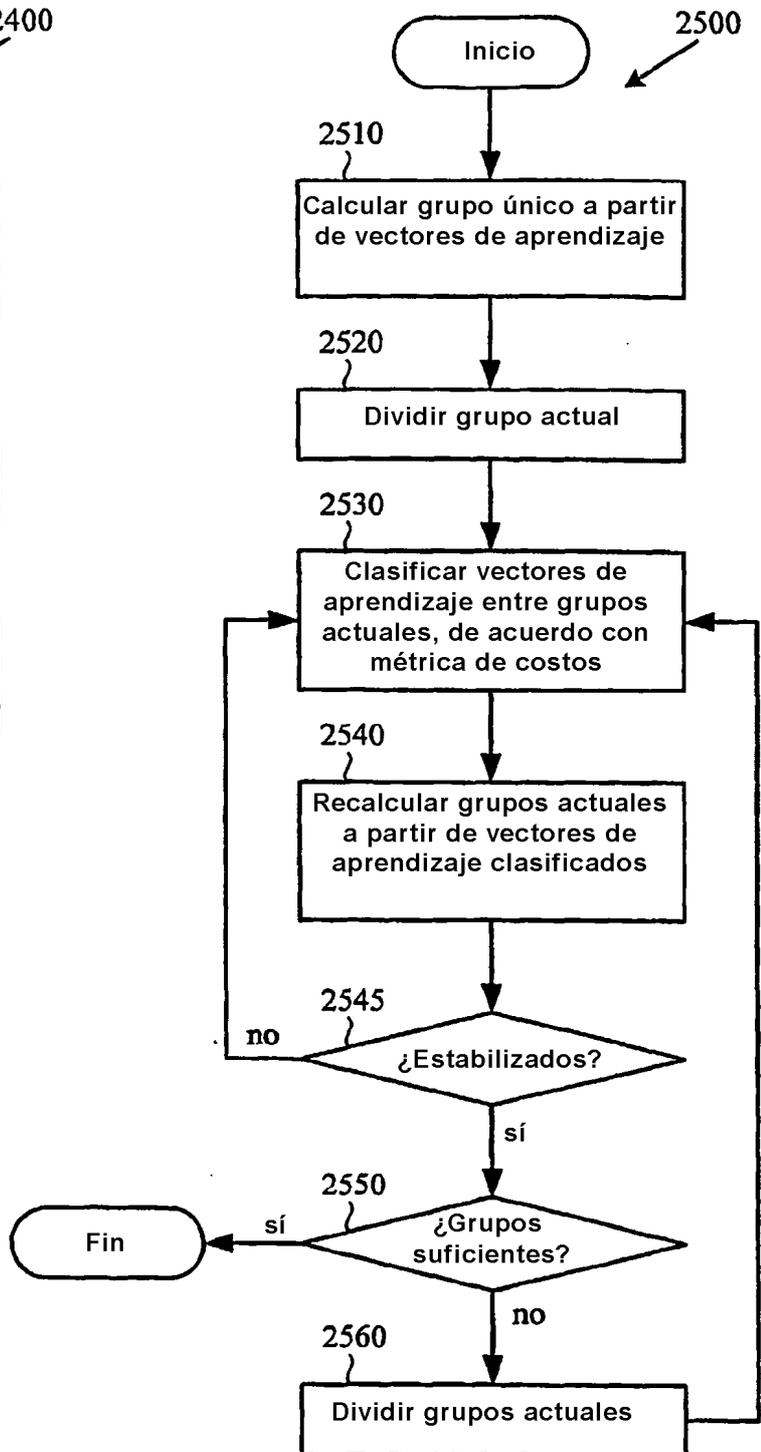


Figura 26

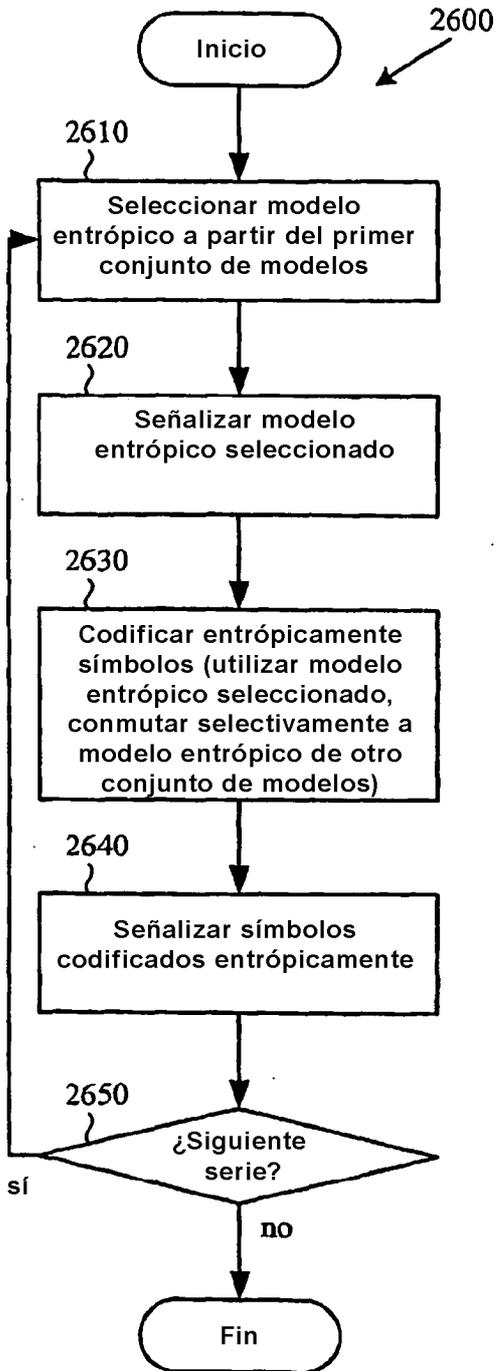


Figura 27

