

19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 727 131**

51 Int. Cl.:

G10L 19/00 (2013.01)

G10L 19/04 (2013.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

96 Fecha de presentación y número de la solicitud europea: **08.02.2012 E 14196260 (5)**

97 Fecha y número de publicación de la concesión europea: **17.04.2019 EP 2863389**

54 Título: **Decodificador con filtros configurables**

30 Prioridad:

16.02.2011 US 201161443360 P

45 Fecha de publicación y mención en BOPI de la traducción de la patente:

14.10.2019

73 Titular/es:

DOLBY LABORATORIES LICENSING CORPORATION (100.0%)

**1275 Market Street
San Francisco, CA 94103, US**

72 Inventor/es:

DAVIS, MARK F.

74 Agente/Representante:

ELZABURU, S.L.P

ES 2 727 131 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

Decodificador con filtros configurables

Referencia cruzada a aplicaciones relacionadas

5 Esta solicitud reclama prioridad a la Solicitud de Patente Provisional de EE.UU. Nº 61/443.360 presentada el 16 de Febrero de 2011.

Campo técnico

10 La invención se refiere a un decodificador. La descripción también describe, como ejemplos útiles para entender la invención, métodos y sistemas para configurar (incluyendo por actualización adaptativa) un filtro de predicción (p. ej., un filtro de predicción en un codificador o decodificador de datos de audio). Ejemplos útiles para entender la invención son métodos y sistemas para generar una paleta de coeficientes de filtro de retroalimentación, y utilizar la paleta para configurar (p. ej., actualizar de manera adaptativa) un filtro de retroalimentación que es (o es un elemento de) un filtro de predicción (p. ej., un filtro de predicción en un codificador o decodificador de datos de audio).

Antecedentes

15 A lo largo de esta descripción, incluyendo en las reivindicaciones, la expresión realizar una operación (p. ej., filtrar o transformar) en señales o datos se utiliza en un amplio sentido para denotar la realización de la operación directamente en las señales o datos, o en versiones procesadas de las señales o datos (p. ej., en versiones de las señales que se han sometido a un filtrado preliminar antes de realizar la operación sobre ellas).

20 A lo largo de esta descripción, incluyendo en las reivindicaciones, la expresión "sistema" se utiliza en un amplio sentido para denotar un dispositivo, sistema, o subsistema. Por ejemplo, un subsistema que predice una próxima muestra en una secuencia de muestra puede ser referido como un sistema de predicción (o predictor), y un sistema que incluye dicho subsistema (p. ej., un procesador que incluye un predictor que predice una próxima muestra en una secuencia de muestras, y medios para utilizar las muestras previstas para realizar la codificación u otro filtrado) también puede ser referido como un sistema de predicción o predictor.

25 A lo largo de esta descripción, incluyendo en las reivindicaciones, el verbo "incluye" se utiliza en un amplio sentido para denotar "que es o incluye" y otras formas del verbo "incluir" se utilizan en el mismo amplio sentido. Por ejemplo, la expresión "un filtro de predicción que incluye un filtro de retroalimentación" (o la expresión "un filtro de predicción incluyendo un filtro de retroalimentación") denota en la presente memoria tanto un filtro de predicción que es un filtro de retroalimentación (es decir, no incluye un filtro de compensación), como un filtro de predicción que incluye un filtro de retroalimentación (y al menos otro filtro, p. ej., un filtro de compensación).

30 Un predictor es un elemento de procesamiento de señal (p. ej., una etapa) utilizado para derivar una estimación de una señal de entrada (p. ej., una muestra actual de un flujo de muestras de entrada) a partir de alguna otra señal (p. ej., muestras en el flujo de muestras de entrada distintas de la muestra actual) y también de manera opcional filtrar la señal de entrada utilizando la estimación. Los predictores a veces se implementan como filtros, generalmente con coeficientes variables en el tiempo que responden a variaciones en las estadísticas de la señal. Normalmente, la salida de un predictor es indicativa de cierta medida de la diferencia entre las señales estimada y original.

35 Una configuración de predictor común encontrada en sistemas de procesamiento de señal digital (DSP) utiliza una secuencia de muestras de una señal objetivo (una señal que se introduce en el predictor) para estimar o predecir una próxima muestra en la secuencia. La intención suele ser reducir la amplitud de la señal objetivo restando cada componente previsto a partir de la muestra correspondiente de la señal objetivo (generando así una secuencia de residuos), y también normalmente codificar la secuencia de residuos resultante. Esto es deseable en sistemas códec de compresión de velocidad de datos, ya que la velocidad de datos requerida normalmente disminuye con el nivel de señal decreciente. El decodificador recupera la señal original a partir de los residuos transmitidos (que pueden ser residuos codificados) realizando cualquier decodificación preliminar necesaria sobre los residuos, y luego replicar el filtrado productivo utilizado por el codificador, y agregar cada valor previsto/estimado al correspondiente de los residuos.

40 A lo largo de esta descripción, incluyendo en las reivindicaciones, la expresión "filtro de predicción" denota tanto un filtro en un predictor como un predictor implementado como un filtro.

45 Cualquier filtro DSP, incluyendo aquellos utilizados en predictores, puede al menos matemáticamente clasificarse como un filtro de compensación (también conocido como un filtro de respuesta de impulso finito o "FIR"), un filtro de retroalimentación (también conocido como filtro de respuesta de impulso infinito o "IIR"), o una combinación de filtros IIR y FIR. Cada tipo de filtro (IIR y FIR) tiene características que lo pueden hacer más susceptible a una u otra aplicación o condición de señal.

50 Los coeficientes de un filtro de predicción deben actualizarse según sea necesario en respuesta a dinámicas de señal para proporcionar estimaciones precisas. En la práctica, esto impone la necesidad de poder calcular de forma rápida

y sencilla coeficientes de filtro aceptables (u óptimos) a partir de la señal de entrada. Existen algoritmos apropiados para filtros de predicción de compensación, como el método de recursión de Levinson-Durbin, pero no existen algoritmos equivalentes para predictores de retroalimentación. Por esta razón, los predictores más prácticos emplean solo la arquitectura de compensación, incluso cuando las condiciones de señal podrían favorecer el uso de una disposición de retroalimentación.

La Patente de EE.UU. 6.664.913, emitida el 16 de Diciembre de 2003 y cedida al cesionario de la presente invención, describe un codificador y un decodificador para decodificar la salida del codificador.

Tanto el codificador como el decodificador incluyen un filtro de predicción. En una clase ejemplos de antecedentes de la técnica (p. ej., el ejemplo mostrado en la FIG. 2 de la presente descripción), el filtro de predicción incluye tanto un filtro IIR como un filtro FIR y se diseña para su uso en la codificación de datos indicativos de una señal en forma de onda (p. ej., una señal de audio o video). En la FIG. 2, el filtro de predicción incluye el filtro FIR 57 (conectado en la configuración de retroalimentación mostrada en la FIG. 2) y el filtro FIR 59, cuyas salidas se combinan por la etapa de resta 56. Los valores de diferencia salientes de la etapa 56 se cuantifican en la etapa de cuantificación 60. La salida de la etapa 60 se suma con las muestras de entrada ("S") en la etapa de suma 61. En funcionamiento, el predictor de la FIG. 2 puede afirmar (como la salida de la etapa 61) valores residuales (identificados en la FIG. 2 como residuos "R"), cada uno indicativo de una suma de una muestra de entrada ("S") y una versión cuantificada y prevista de dicha muestra (donde dicha versión prevista de la muestra se determina por la diferencia entre las salidas de los filtros 57 y 59).

Codificadores y decodificadores disponibles en el mercado que incorporan la tecnología "Dolby TrueHD", desarrollados por Dolby Laboratories Licensing Corporation, emplean métodos de codificación y decodificación del tipo descrito en la Patente de EE.UU. 6.664.913. Un codificador que incorpora la tecnología Dolby TrueHD es un codificador de audio digital sin pérdidas, lo que significa que la salida decodificada (producida a la salida de un decodificador compatible) debe coincidir exactamente con la entrada del decodificador, bit por bit. Esencialmente, el codificador y el decodificador comparten un protocolo común para expresar ciertas clases de señales de una forma más compacta, de manera que se reduce la velocidad de datos transmitida pero el codificador puede recuperar la señal original.

La Patente de EE.UU. 6.664.913 sugiere que los filtros 57 y 59 (y filtros de predicción similares) pueden configurarse para minimizar la velocidad de datos codificados (la velocidad de datos de la salida "R") probando cada uno de un pequeño conjunto de opciones de coeficientes de filtro posibles (utilizando cada conjunto de prueba para codificar la forma de onda de entrada), seleccionando el conjunto que entrega el nivel de señal de salida promedio más bajo o el nivel de pico más pequeño en un bloque de datos de salida (generado en respuesta a un bloque de datos de entrada), y configurando los filtros con el conjunto de coeficientes seleccionado. La patente sugiere además que el conjunto de coeficientes seleccionado puede transmitirse al codificador, y cargarse en un filtro de predicción en el decodificador para configurar el filtro de predicción.

La Patente de EE.UU. 7.756.498, emitida el 13 de Julio de 2010, describe un terminal de comunicación móvil que se mueve a velocidad variable mientras que recibe una señal. El terminal incluye un predictor que incluye un filtro IIR de primer orden, y se proporciona una lista de parejas predeterminadas de coeficientes de filtro IIR al predictor. Durante el funcionamiento del terminal (mientras que se mueve a una velocidad específica), se selecciona una pareja de coeficientes de filtro IIR predeterminada a partir de la lista de filtros candidatos para configurar el filtro (la selección se basa en la comparación de los resultados de la predicción con los resultados en los que no se produce ruido). La selección puede actualizarse según varíe la velocidad del terminal, pero no hay ninguna sugerencia para abordar la cuestión de la continuidad de la señal ante el cambio de los coeficientes del filtro. La referencia no enseña como se genera la lista de filtros candidatos, excepto para indicar que cada pareja en la lista se determina como resultado de experimentación (no descrita) que es adecuada para configurar el filtro cuando el terminal se mueve a una velocidad diferente.

Aunque se ha propuesto actualizar de manera adaptativa un filtro IIR (p. ej., el filtro 57 en el sistema de la FIG. 2) de un filtro de predicción (p. ej., para minimizar la energía de la señal de salida de momento en momento), no se sabía como hacerlo de manera efectiva, rápida y eficiente (p. ej., para optimizar el filtro IIR, y/o un filtro de predicción que incluye un filtro IIR, para su uso de manera rápida y efectiva bajo las condiciones de señal relevantes, que pueden cambiar con el tiempo). Tampoco se sabía como hacerlo de una manera que aborde la cuestión de la continuidad de la señal bajo la condición de coeficientes de filtro cambiantes.

La Patente de EE.UU. 6.664.913 también sugiere que determinar un primer grupo de posibles conjuntos de coeficientes del filtro de predicción (un pequeño número de conjuntos a partir de los cuales se puede seleccionar un conjunto deseado) para incluir conjuntos que determinan filtros muy diferentes que coinciden con los espectros de forma de onda esperados normalmente. Luego un segundo paso de selección de coeficientes puede realizarse (después de seleccionar uno de los mejores de los conjuntos en el primer grupo) para hacer una selección refinada de un mejor conjunto de coeficientes de filtro a partir de un pequeño segundo grupo de posibles conjuntos de coeficientes del filtro de predicción, donde todos los conjuntos en el segundo grupo determinan filtros similares al filtro seleccionado durante el primer paso. Este proceso puede iterarse, utilizando cada vez un grupo más similar de posibles filtros de predicción que el utilizado en la iteración anterior.

Aunque ha sido propuesto para generar uno o más grupos pequeños de posibles conjuntos de coeficientes del filtro de predicción (a partir de los cuales se puede seleccionar un conjunto de coeficientes deseado para configurar un filtro de predicción), no se sabía como determinar dicho pequeño grupo de manera efectiva y eficiente, de manera que cada conjunto en el grupo es útil para optimizar (o actualizar de manera adaptativa) un filtro IIR (o un filtro de predicción que incluye un filtro IIR) para su uso bajo condiciones de señal relevantes.

Breve descripción de la invención

Según la invención, se proporciona un decodificador como se establece en la reivindicación 1. Realizaciones preferidas se exponen en las reivindicaciones dependientes.

Un ejemplo útil para entender la invención es un método para utilizar una paleta predeterminada de conjuntos de coeficientes de filtro IIR (retroalimentación) para configurar (p. ej., actualizar de manera adaptativa) un filtro IIR que es (o es un elemento de) un filtro de predicción. Normalmente, el filtro de predicción se incluye en un sistema de codificación de datos de audio (codificador) o en un sistema de decodificación de datos de audio (decodificador). En ejemplos típicos útiles para entender la invención, el método utiliza una paleta predeterminada de conjuntos de coeficientes de filtro IIR ("conjuntos de coeficientes IIR") para configurar un filtro de predicción que incluye tanto un filtro IIR como un filtro FIR (compensación), y el método incluye los pasos de: para cada uno de los conjuntos de coeficientes IIR en la paleta, generar datos de configuración indicativos de la salida generada aplicando el filtro IIR configurado con cada uno de los conjuntos de coeficientes IIR a los datos de entrada, e identificar (como un conjunto de coeficientes IIR seleccionado) uno de los conjuntos de coeficientes IIR que configura el filtro IIR para generar datos de configuración que tienen un nivel más bajo (p. ej., el nivel RMS más bajo) o que configura el filtro IIR para cumplir una combinación óptima de criterios (incluyendo el criterio de que los datos de configuración tienen un nivel más bajo); luego determinar un conjunto de coeficientes de filtro IIR óptimo realizando una operación de recursión (p. ej., recursión de Levinson-Durbin) en datos de prueba indicativos de la salida generada aplicando el filtro de predicción a los datos de entrada con el filtro IIR configurado con el conjunto de coeficientes IIR seleccionado (normalmente, se emplea un conjunto predeterminado de coeficientes de filtro FIR como un conjunto inicial de coeficientes FIR candidato para la recursión, y se emplean otros conjuntos candidatos de coeficientes de filtro FIR en iteraciones sucesivas de la operación de recursión hasta que la recursión converge para determinar el conjunto de coeficientes de filtro FIR óptimo), y configurar el filtro FIR con el conjunto de coeficientes FIR óptimo y configurar el filtro IIR con el conjunto de coeficientes IIR seleccionado, configurando así el filtro de predicción.

Cuando el filtro de predicción se incluye en un codificador y se ha configurado, el codificador puede operarse para generar datos de salida codificados codificando datos de entrada (con el filtro de predicción normalmente generando valores residuales que se emplean para generar los datos de salida codificados), y los datos de salida codificados pueden afirmarse (p. ej., a un decodificador o a un medio de almacenamiento para su posterior provisión a un decodificador) con datos de coeficientes de filtro indicativos del conjunto de coeficientes IIR seleccionado (con el que el filtro IIR fue configurado durante la generación de los datos de salida codificados). Los datos de los coeficientes del filtro son normalmente el propio conjunto de coeficientes IIR seleccionado, pero alternativamente podrían ser datos (p. ej., un índice a una paleta o tabla de consulta) indicativos del conjunto de coeficientes IIR seleccionado.

En algunos ejemplos útiles para entender la invención, se identifica el conjunto de coeficientes IIR seleccionado (el conjunto de coeficientes en la paleta que se selecciona para configurar el filtro IIR) como el conjunto de coeficientes IIR en la paleta que configura el filtro IIR para generar datos de salida (en respuesta a datos de entrada) que tienen un nivel más bajo de $A + B$, donde "A" es el nivel (p. ej., el nivel RMS) de los datos de salida y "B" es la cantidad de datos de cadena lateral necesaria para identificar el conjunto de coeficientes IIR (p. ej., la cantidad de datos de cadena lateral que debe transmitirse a un decodificador para permitir al decodificador identificar el conjunto de coeficientes IIR) y también, de manera opcional, de cualesquiera otros datos de cadena lateral requeridos para decodificar los datos que han sido codificados utilizando el filtro de predicción configurado con el conjunto de coeficientes IIR. Este criterio es apropiado en algunos ejemplos útiles para entender la invención dado que algunos de los conjuntos de coeficientes IIR en la paleta pueden comprender coeficientes más largos (más precisos) que otros, de manera que un filtro IIR menos efectivo (considerando solo RMS de los datos de salida) determinado por coeficientes cortos puede elegirse sobre un filtro IIR ligeramente más efectivo determinado por coeficientes más largos.

En algunos ejemplos útiles para entender la invención, la sincronización (p. ej., frecuencia) con la que se realiza o permite que se realice la actualización adaptativa de la configuración de un filtro de predicción (que incluye un filtro IIR, o un filtro IIR y un filtro FIR) está restringida (p. ej., para optimizar la eficiencia de la codificación de predicción). Por ejemplo, cada vez que se reconfigura un filtro de predicción de un codificador sin pérdidas típico, hay un cambio de estado en el codificador que puede requerir que los datos de sobrecarga (datos de cadena lateral) indicativos del nuevo estado se transmitan para permitir que un decodificador tenga en cuenta cada cambio de estado durante la decodificación. Sin embargo, si el cambio de estado del codificador sucede por alguna razón que no es una reconfiguración del filtro de predicción (p. ej., un cambio de estado que ocurre tras el comienzo del procesamiento de un nuevo bloque, p. ej., macro-bloque de muestras), los datos de sobrecarga indicativos del nuevo deben también transmitirse al decodificador de manera que una reconfiguración del filtro de predicción puede realizarse en este momento sin agregar (o sin agregar de una manera significativa o intolerable) a la cantidad de sobrecarga que debe transmitirse. En algunos ejemplos útiles para entender la invención, se realiza una operación de determinación de continuidad para determinar cuando hay un cambio de estado del codificador, y la sincronización de las operaciones

de reconfiguración del filtro de predicción se controla en consecuencia (p. ej., la reconfiguración del filtro de predicción se aplaza hasta que se produce un evento de cambio de estado).

En otra clase de ejemplos útiles para entender la invención, el ejemplo es un método para generar una paleta predeterminada de coeficientes de filtro IIR que pueden utilizarse para configurar (p. ej., actualizar de manera adaptativa) un filtro de predicción IIR ("retroalimentación") (es decir, un filtro IIR que es o es un elemento de un filtro de predicción). La paleta comprende al menos dos conjuntos (normalmente un número de conjuntos pequeño) de coeficientes de filtro IIR, cada uno de los conjuntos consta de coeficientes suficientes para configurar el filtro IIR. En una clase de ejemplos útiles para entender la invención, cada conjunto de coeficientes en la paleta se genera realizando optimización no lineal sobre un conjunto (un "conjunto de entrenamiento") de señales de entrada, sujeto al menos a una restricción. Normalmente, la optimización se realiza sujeta a múltiples restricciones, incluyendo al menos dos de la mejor predicción, Q de filtro máxima, zumbido, precisión numérica permitida o requerida de los coeficientes del filtro (p. ej., el requerimiento de que cada coeficiente en un conjunto debe consistir en no más de X bits, donde X puede ser igual a 14 bits por ejemplo), sobrecarga de transmisión, y restricciones de estabilidad del filtro. Se aplica al menos un algoritmo de optimización no lineal (p. ej., optimización Newtoniana y/o optimización Simple) para cada bloque de cada señal en el conjunto de entrenamiento, para llegar a un conjunto óptimo candidato de coeficientes de filtro para la señal. Se agrega el conjunto óptimo candidato a la paleta si el filtro IIR determinado así satisface cada restricción, pero se rechaza (y no se agrega a la paleta) si el filtro IIR viola al menos una restricción (p. ej., el filtro IIR no es estable). Si se rechaza un conjunto óptimo candidato, un conjunto candidato (determinado por la misma optimización en la misma señal) igualmente bueno (o el siguiente mejor) puede añadirse a la paleta si el conjunto candidato igualmente bueno (o el siguiente mejor) satisface cada restricción, y el proceso se itera hasta que un conjunto de coeficientes (determinado a partir de la señal) se haya añadido a la paleta. La paleta puede incluir conjuntos de coeficientes de filtro determinados utilizando diferentes algoritmos de optimización restringida (p. ej., optimización Newtoniana restringida y optimización Simple restringida pueden realizarse por separado, y las mejores soluciones de cada una se seleccionan para su inclusión en la paleta). Si la optimización restringida produce una paleta inicial inaceptablemente grande, se emplea un proceso de poda para reducir el tamaño de la paleta (borrando al menos un conjunto de la paleta inicial), basándose en una combinación de la acumulación del histograma y de la mejora neta proporcionada por cada conjunto de coeficientes en la paleta inicial sobre las señales en el conjunto de entrenamiento.

Preferiblemente, la paleta de conjuntos de coeficientes de filtro IIR se determina de manera que incluye conjuntos de coeficientes que configurarán óptimamente un filtro de predicción IIR para su uso con cualquier señal de entrada que tiene características en un rango esperado.

Aspectos de los ejemplos útiles para entender la invención incluyen un sistema (p. ej., un codificador o un sistema que incluye tanto un codificador como un decodificador) configurado (p. ej., programado) para realizar cualquier método descrito, y un medio legible por ordenador (p. ej., un disco) que almacena el código para programar un procesador u otro sistema para realizar cualquier método descrito.

Breve descripción de los dibujos

La FIG. 1 es un diagrama de bloques de un codificador que incluye un filtro de predicción que incluye un filtro IIR (7) y un filtro FIR (9). El filtro de predicción se configura (y se actualiza de manera adaptativa) utilizando una paleta predeterminada (8) de conjuntos de coeficientes IIR.

La FIG. 2 es un diagrama de bloques de un filtro de predicción, de un tipo empleado en un codificador convencional, que incluye un filtro IIR y un filtro FIR. La FIG. 3 es un diagrama de bloques de un decodificador configurado para decodificar datos que han sido codificados por el codificador de la FIG. 1. El decodificador de la FIG. 3 incluye un filtro IIR que se configura (y se adapta de manera adaptativa) de acuerdo con una realización de la invención.

La FIG. 4 es una vista en alzado de un disco óptico legible por ordenador en el que se almacena el código.

Descripción detallada de las realizaciones preferidas

Muchas realizaciones de la presente invención son tecnológicamente posibles. Será evidente para los expertos en la técnica a partir de la presente descripción como implementarlas. Se describirán realizaciones del decodificador de la invención con referencia a la Fig. 3.

En una realización típica, el sistema de la FIG. 3 se implementa como un procesador de señal digital (DSP) cuya arquitectura es adecuada para procesar los datos de entrada esperados y que se configura (p. ej., se programa) con firmware y/o software apropiado para implementar una realización del método de la invención. El DSP podría implementarse como un circuito integrado (o conjunto de chips) e incluiría el programa y la memoria de datos accesibles por su procesador (es). La memoria incluiría memoria no volátil adecuada para almacenar la paleta de coeficientes de filtro, los datos del programa, y otros datos requeridos para implementar cada realización del método de la invención a realizar. Alternativamente, el sistema de la FIG. 3 se implementa como un procesador de propósito general programado con software apropiado para implementar una realización del método de la invención, o se implementa en hardware configurado apropiadamente.

Normalmente, múltiples canales de muestras de datos de entrada se afirman a las entradas del codificador 1 (de la FIG. 1). Cada canal normalmente incluye un flujo de muestras de audio de entrada y puede corresponder a un canal diferente de un programa de audio de canal múltiple. En cada canal, el codificador 1 normalmente recibe bloques relativamente pequeños (“micro-bloques”) de muestras de audio de entrada. Cada micro-bloque puede constar de 48 muestras.

El decodificador 1 se configura para realizar las siguientes funciones: una operación de re-matrización (representado por la etapa de re-matrización 3 de la FIG. 1), una operación de predicción (que incluye la generación de muestras previstas y la generación de residuos de las mismas) representada por el predictor 5, una operación de codificación de representación de punto flotante de bloque (representada por la etapa 11), una operación de codificación de Huffman (representada por la etapa de codificación de Huffman 13), y una operación de empaquetado (representada por la etapa de empaquetado 15). En algunas implementaciones, el codificador 1 es un procesador de señal digital (DSP) programado y configurado de otra manera para realizar estas funciones (y opcionalmente funciones adicionales) en software.

La etapa de re-matrización 3 codifica las muestras de audio de entrada (para reducir su tamaño/nivel de una manera reversible), generando así muestras codificadas. En implementaciones típicas en las que se ingresan múltiples canales de muestras de entrada a la etapa de re-matrización 3 (p. ej., cada una correspondiente a un canal de un programa de audio de canal múltiple), la etapa 3 determina si generar una suma o una diferencia de muestras de cada una de al menos una pareja de los canales de entrada, y saca bien los valores de suma y de diferencia (p. ej., una versión ponderada de cada suma o diferencia) o bien, las propias muestras de entrada, con datos de cadena lateral que indican si se están sacando los valores de suma y de diferencia o las propias muestras de entrada. Normalmente, los valores de suma y de diferencia salientes de la etapa 3 son sumas y diferencias ponderadas de muestras, y los datos de cadena lateral incluyen los coeficientes de suma/diferencia. El proceso de re-matrización realizado en la etapa 3 forma sumas y diferencias de señales del canal de entrada para cancelar componentes de señal duplicados. Por ejemplo, dos canales de 16 bits idénticos podrían codificarse (en la etapa 3) como una señal de suma de 17 bits y una señal de silencio de diferencia, para lograr un ahorro potencial de 15 bits por muestra, menos cualquier información de cadena lateral necesaria para revertir la re-matrización en el decodificador.

Por conveniencia, la siguiente descripción de las operaciones posteriores realizadas en el codificador 1 se refiere a muestras (y su codificación) en uno solo de los canales representados por la salida de la etapa 3. Se entenderá que la codificación descrita se realiza en las muestras (identificadas en la FIG. 1 como muestras “ S_x ”) en todos los canales.

El predictor 5 realiza las siguientes operaciones: restar (representado por la etapa de resta 4 y la etapa de resta 6), filtrado IIR (representado por el filtro IIR 7), filtrado FIR (representado por el filtro FIR 9), cuantificación (representada por la etapa de cuantificación 10), configuración del filtro IIR 7 (para implementar conjuntos de coeficientes IIR seleccionados a partir de la paleta de coeficientes IIR 8), configuración del filtro FIR 9, y actualización adaptativa de las configuraciones de los filtros 7 y 9. En respuesta a la secuencia de muestras codificadas (re-matrizadas) generada en la etapa 3, el predictor 5 predice cada “próxima” muestra codificada en la secuencia. Los filtros 7 y 9 se implementan de manera que sus salidas combinadas (en respuesta a la secuencia de muestras codificadas de la etapa 3) son indicativas de una próxima muestra codificada prevista en la secuencia. Las próximas muestras codificadas previstas (generadas en la etapa 6 restando la salida del filtro 7 de la salida del filtro 9) se cuantifican en la etapa 10. Específicamente, en la etapa de cuantificación 10, se realiza una operación de redondeo (p. ej., al entero más cercano) en cada próxima muestra codificada prevista generada en la etapa 6.

En la etapa 4, el predictor 5 resta cada valor actual de la salida cuantificada, y combinada, P_n , de los filtros 7 y 9 de cada valor actual de la secuencia de muestras codificadas de la etapa 3 para generar una secuencia de valores residuales (residuos). Los valores residuales son indicativos de la diferencia entre cada muestra codificada de la etapa 3 y una versión prevista de dicha muestra codificada. Los valores residuales generados en la etapa 4 se afirman a la etapa de representación de punto flotante de bloque 11.

Más específicamente, en la etapa 4 la salida cuantificada, y prevista, P_n , de los filtros 7 y 9 (en respuesta a muestras anteriores, que incluyen la “(n-enésima-1)” muestra codificada, de la secuencia de muestras codificadas de la etapa 3 y la secuencia de valores residuales de la etapa 4) se resta de la “(n)”-enésima muestra codificada de la secuencia para generar el residuo “(n)”, donde P_n es una versión cuantificada de la diferencia $Y_n - X_n$, donde X_n es el valor actual afirmado a la salida del filtro 7 en respuesta a los valores residuales anteriores, Y_n es el valor actual afirmado a la salida del filtro 9 en respuesta a las muestras codificadas anteriores en la secuencia, y $Y_n - X_n$ es la “(n)”-enésima muestra codificada prevista en la secuencia.

Antes de la operación del filtro IIR 7 y del filtro FIR 9 para filtrar las muestras codificadas generadas en la etapa 3, el predictor 5 realiza una operación de selección de coeficientes IIR (a describir a continuación) para seleccionar un conjunto de coeficientes de filtro IIR (de aquellos conjuntos predeterminados almacenados previamente en la paleta de coeficientes IIR 8), y configura el filtro IIR 7 para implementar el conjunto seleccionado de coeficientes IIR en el mismo. El predictor 5 también determina coeficientes de filtro FIR para configurar el filtro FIR 9 para su funcionamiento con el filtro IIR 7 así configurado. La configuración de los filtros 7 y 9 se actualiza de manera adaptativa de un modo a describir. El predictor 5 también afirma a la etapa de empaquetado 15 datos de “coeficientes de filtro” indicativos del conjunto de coeficientes de filtro IIR seleccionado actualmente (a partir de la paleta 8), y también opcionalmente, del

conjunto actual de coeficientes de filtro FIR. En algunas implementaciones, los datos de “coeficientes de filtro” son el conjunto de coeficientes de filtro IIR seleccionado actualmente (y también opcionalmente, el conjunto de coeficientes de filtro FIR actual correspondiente). Alternativamente, los datos de coeficientes de filtro son indicativos del conjunto de coeficientes IIR (o FIR e IIR) seleccionado actualmente. La paleta 8 puede implementarse como una memoria del codificador 1, o como ubicaciones de almacenaje en una memoria del codificador 1, en la que se ha almacenado previamente un número de conjuntos de coeficientes de filtro IIR predeterminados diferentes (de manera que sea accesible por el predictor 5 para configurar el filtro 7 y para actualizar la configuración del filtro 7).

En relación con la actualización adaptativa de las configuraciones de los filtros 7 y 9, el predictor 5 es operable preferiblemente para determinar cuántos micro-bloques de las muestras codificadas (generadas en la etapa 3) para codificar aún más utilizando cada configuración determinada de los filtros 7 y 9. En efecto, el predictor 5 determina el tamaño de un “macro-bloque” de las muestras codificadas que se codificará utilizando cada configuración determinada de los filtros 7 y 9 (antes de que la configuración se actualice). Por ejemplo, una realización preferida del predictor 5 puede determinar un número N (donde N está en el rango de $1 \leq N \leq 128$) de micro-bloques para codificar utilizando cada configuración determinada de los filtros 7 y 9. La configuración (y actualización adaptativa) de los filtros 7 y 9 se describirá con más detalle a continuación.

La etapa de representación de punto flotante de bloque 11 opera sobre los residuos cuantificados generados en la etapa de predicción 5 y sobre las palabras de cadena lateral (“datos MSB”) también generadas en la etapa de predicción 5. Los datos MSB son indicativos de los bits más significativos (MSBs) de las muestras codificadas que corresponden a los residuos cuantificados determinados en la etapa de predicción 5. Cada uno de los residuos cuantificados es en sí mismo indicativo de solo los bits menos significativos de una diferente de las muestras codificadas. Los datos MSB pueden ser indicativos de los bits más significativos (MSBs) de la muestra codificada correspondiente al primer residuo cuantificado de cada macro-bloque determinado en la etapa de predicción 5.

En la etapa de representación de punto flotante de bloque 11, los bloques de residuos cuantificados y los datos MSB generados en el predictor 5 se codifican aún más. Específicamente, la etapa 11 genera datos indicativos de un exponente maestro para cada bloque, y de las mantisas individuales para los residuos cuantificados individuales en cada bloque.

Se utilizan cuatro procesos de codificación clave en el codificador 1 de la FIG. 1: re-matrización, predicción, codificación de Huffman, y representación de punto flotante de bloque. El proceso de representación de punto flotante de bloque (implementado por la etapa 11) se implementa preferiblemente para aprovechar el hecho de que las señales silenciosas se pueden transmitir de manera más compacta que las señales ruidosas. Un bloque indicativo de una señal de 16 bits de nivel completa, por ejemplo, que se ingresa a la etapa 11 puede requerir que se transporten los 16 bits de cada muestra (es decir, salida de la etapa 11). Sin embargo, un bloque de valores indicativo de una señal 48 dB más baja en nivel (que se afirma a la entrada de la etapa 11) solo requerirá que 8 bits por muestra salgan de la etapa 11, junto con una palabra de cadena lateral que indica que los 8 bits superiores de cada muestra están sin ejercer y suprimidos (y necesita ser restaurada por el decodificador).

En el sistema de la FIG. 1, el objetivo de la re-matrización (en la etapa 3) y de la codificación de predicción (en el predictor 5) es reducir el nivel de señal lo máximo posible, de una manera reversible, para obtener el máximo beneficio de la codificación de punto flotante de bloque en la etapa 11.

Los valores codificados generados durante la etapa 11 sometidos a la codificación de Huffman en la etapa del codificador de Huffman 13 para reducir aún más su tamaño/nivel de una manera reversible. Los valores codificados de Huffman resultantes se empaquetan (con datos de cadena lateral) en la etapa de empaquetado 15 para salir del codificador 1. La etapa del codificador de Huffman 13 reduce preferiblemente el nivel de muestras individuales que ocurren comúnmente sustituyendo cada palabra de código más corta a partir de una tabla de búsqueda (cuya inversa se implementa en el decodificador de Huffman 25 del sistema de la FIG. 3), permitiendo la restauración de la muestra original mediante la búsqueda inversa de tabla en el decodificador de la FIG. 3.

En la etapa de empaquetado 15 se genera un flujo de datos de salida empaquetando juntos los valores codificados de Huffman (del codificador 13), las palabras de cadena lateral (recibidas de cada etapa del codificador 1 en la que son generados), y los datos de coeficientes de filtro (del predictor 5) que determinan la configuración actual del filtro IIR 7 (y también normalmente la configuración actual del filtro FIR 9). El flujo de datos de salida son datos codificados (indicativos de las muestras de audio de entrada) que son datos comprimidos (ya que la codificación realizada en el codificador 1 es compresión sin pérdidas). En un decodificador (p. ej., el decodificador 21 de la FIG. 3), el flujo de datos de salida puede decodificarse para recuperar las muestras de audio de entrada originales sin pérdidas.

En ejemplos alternativos útiles para entender la invención, el filtro de predicción de la etapa del predictor 5 se implementa para tener una estructura diferente a la que mostrada en la FIG. 1 (p. ej., la estructura de cualquiera de las realizaciones descritas en la Patente de EE.UU. 6.664.913 citada anteriormente), pero se puede configurar (p. ej., se puede actualizar de manera adaptativa) utilizando una paleta de coeficientes IIR predeterminada. El filtro de predicción de la etapa del predictor 5 se puede implementar (con la estructura mostrada en la FIG. 1) de una manera convencional (p. ej., como se describe en la Patente de EE.UU. 6.664.913 citada anteriormente), excepto que la implementación convencional se modifica de modo que el filtro de predicción se puede configurar (y se puede

actualizar de manera adaptativa) utilizando un paleta de coeficientes IIR predeterminada (paleta 8). Durante dicha actualización, se selecciona y emplea un conjunto de coeficientes de filtro IIR (de los incluidos en la paleta 8) para el filtro IIR 7, y se configura el filtro FIR 9 para funcionar aceptablemente (u óptimamente) con el filtro 7 así configurado. El filtro FIR 9 puede ser idéntico al filtro FIR 59 de la FIG. 2, excepto en que cada valor de salida de dicha implementación de filtro 9 es el inverso aditivo del valor que se emitiría desde el filtro 59 en respuesta a la misma entrada, la etapa de resta 6 (del predictor 5 de la FIG. 1) puede reemplazar la etapa de resta 56 de la FIG. 2, la etapa de resta 4 (del predictor 5 de la FIG. 1) puede reemplazar la etapa de suma 61 de la FIG. 2, y el filtro IIR 7 (del predictor 5 de la FIG. 1) puede ser idéntico al filtro FIR 57 de la FIG. 2 (conectado en la configuración de retroalimentación mostrada en la FIG. 2), excepto en que cada valor saliente de dicha implementación de filtro 7 es el inverso aditivo del valor que se emitiría desde el filtro 57 en respuesta a la misma entrada.

A continuación describimos el decodificador 21 de la FIG. 3.

Normalmente, múltiples canales de muestras de datos de entrada codificadas se afirman a las entradas del decodificador 21. Cada canal normalmente incluye un flujo de muestras de audio de entrada codificadas y puede corresponder a un canal diferente (o mezcla de canales determinada por la re-matrización en el codificador 1) de un programa de audio de canal múltiple.

El decodificador 21 se configura para realizar las siguientes funciones: una operación de desempaquetado (representada por la etapa de desempaquetado 23 de la FIG. 3), una operación de decodificación de Huffman (representada por la etapa de decodificación de Huffman 25), una operación de decodificación de representación de punto flotante de bloque (representada por la etapa 27), una operación de predicción (que incluye la generación de muestras previstas y la generación de muestras previstas a partir de ellas) representada por el predictor 29, y una operación de re-matrización (representada por la etapa de re-matrización 41). En algunas implementaciones, el decodificador 21 es un procesador de señal digital (DSP) programado y configurado de otra manera para realizar estas funciones (y opcionalmente funciones adicionales) en software.

El decodificador 21 funciona como sigue: la etapa de desempaquetado 23 desempaqueta los valores codificados de Huffman (del codificador 13 del codificador 1), todas las palabras de cadena lateral (de las etapas del codificador 1), y los datos de coeficientes de filtro (del predictor 5 del codificador 1), y proporciona los valores codificados desempaquetados para procesar en el decodificador de Huffman 25, los datos de coeficientes de filtro para procesar en el predictor 29, y subconjuntos de palabras de cadena lateral para procesar en las etapas del decodificador 21 según sea apropiado. La etapa 23 desempaqueta valores que determinan el tamaño (p. ej., número de micro-bloques) de cada macro-bloque de valores codificados de Huffman recibidos (el tamaño de cada macro-bloque determinaría los intervalos en los que deben reconfigurarse el filtro IIR 31 y el filtro FIR 33 (del predictor 29 del decodificador 21)).

En la etapa de decodificación de Huffman 25, los valores codificados de Huffman se decodifican (realizando la inversa de la operación de codificación de Huffman realizada en el codificador 1), y los valores decodificados de Huffman resultantes se proporcionan a la etapa de decodificación de representación de punto flotante de bloque 27.

En la etapa de decodificación de representación de punto flotante de bloque 27, se realiza la inversa de la operación de codificación que fue realizada en la etapa 11 del codificador 1 (en bloques de los valores decodificados de Huffman) para recuperar los valores codificados V_x . Cada uno de los valores V_x es igual a la suma de un residuo cuantificado que fue generado por el predictor del codificador (cada residuo cuantificado corresponde a una muestra codificada, S_x , generada en la etapa de re-matrización 3 del codificador 1) y los MSBs de la muestra codificada, S_x . El valor del residuo cuantificado es $S_x - P_x$, donde P_x es el valor previsto de S_x , generado en el predictor 5 del codificador 1. Los valores codificados V_x se proporcionan a la etapa del predictor 29. En efecto, cada exponente determinado por la salida de la etapa de punto flotante de bloque 11 del codificador 1 se agrega de nuevo a las mantisas del bloque relevante (también determinado por la salida de la etapa 11). El predictor 29 opera sobre el resultado de esta operación.

En el predictor 29, el filtro FIR 33 normalmente es idéntico al filtro IIR 7 del codificador 1 de la FIG. 1, excepto en que el filtro FIR 33 se conecta en una configuración de compensación en el predictor 29 (mientras que el filtro 7 se conecta en una configuración de retroalimentación en el predictor 5 del codificador 1), y el filtro IIR 31 normalmente es idéntico al filtro FIR 9 del codificador 1 de la FIG. 1, excepto en que el filtro IIR 31 se conecta en una configuración de retroalimentación en el predictor 29 (mientras que el filtro 9 se conecta en una configuración de compensación en el predictor 5 del codificador 1). En dichas realizaciones típicas, cada uno de los filtros 7, 9, 31, y 33 se implementa con una estructura de filtro FIR (y cada uno puede ser considerado como un filtro FIR), pero cada uno de los filtros 7 y 31 es referido en la presente memoria como un filtro "IIR" cuando se conecta en una configuración de retroalimentación.

El predictor 29 realiza las siguientes operaciones: resta (representada por la etapa de resta 30), suma (representada por la etapa de suma 34), filtrado IIR (representada por el filtro IIR 31), filtrado FIR (representada por el filtro FIR 33), cuantificación (representada por la etapa de cuantificación 32), y configuración del filtro IIR 31 y del filtro FIR 33, y actualización de las configuraciones de los filtros 31 y 33. En respuesta a los datos de coeficientes de filtro (del predictor 5 del codificador, como desempaquetado en la etapa 23), el predictor 29 configura el filtro FIR 33 con uno seleccionado de los conjuntos de coeficientes IIR de la paleta de coeficientes IIR 8 (este conjunto de coeficientes

normalmente es idéntico a un conjunto de coeficientes que fue empleado en el codificador 1 para configurar el filtro IIR 7), y normalmente también configura el filtro IIR 31 con los coeficientes incluidos en (o de otra manera determinados por) los datos de coeficientes de filtro (estos coeficientes normalmente son idénticos a los coeficientes que fueron empleados en el codificador 1 para configurar el filtro FIR 9). Si los datos de coeficientes de filtro determinan (en lugar de incluir) el conjunto actual de coeficientes IIR que se utilizará para configurar el filtro 33, el conjunto actual de coeficientes IIR se carga a partir de la paleta 8 del predictor 29 (en la FIG. 3) en el filtro 33 (es este caso, la paleta 8 de la FIG. 3 es idéntica a la paleta numerada idénticamente del predictor 5 en la Figura 1).

Si los datos de coeficientes de filtro incluyen (en lugar de determinar) el conjunto actual de coeficientes IIR a utilizar para configurar el filtro 33, entonces se omite la paleta 8 del decodificador 21 (es decir, la paleta de coeficientes IIR no se carga previamente en el decodificador 21) y los propios datos de coeficientes de filtro se utilizan para configurar el filtro 33. Como se ha señalado, el realizaciones alternativas en las que los datos de coeficientes de filtro determinan uno de los conjuntos de coeficientes IIR (en la paleta 8) a utilizar para configurar el filtro 33, entonces este conjunto de coeficientes IIR se puede seleccionar de la paleta 8 (que ha sido cargada previamente en el decodificador 21) y utilizar para configurar el filtro 33. En cualquier caso, el filtro 33 (cuando se utilizar para decodificar datos que han sido codificados en el predictor 5 con el filtro 7 que utiliza un conjunto específico de coeficientes IIR) se configura con el mismo conjunto de coeficientes IIR. De manera similar, cuando los datos de coeficientes de filtro incluyen un conjunto de coeficientes FIR que ha sido utilizado para configurar el filtro FIR 9 del predictor 5 (de la FIG. 1), el filtro IIR 31 se configura con este conjunto de coeficientes FIR (para su uso por el filtro 31 para decodificar datos que han sido codificados en el predictor 5 con el filtro 9 utilizando los mismos coeficientes FIR). La configuración del filtro FIR 33 (y del filtro IIR 31) se actualiza normalmente en respuesta a cada nuevo conjunto de datos de coeficientes de filtro.

En implementaciones del decodificador alternativas (en las que la paleta 8 de la FIG. 3 normalmente no es idéntica a la paleta 8 de la FIG. 1, pero en las que la paleta 8 de la FIG. 3 incluye conjuntos predeterminados de coeficientes IIR para configurar el filtro 31), el predictor 29 es operable en un modo de configuración (p. ej., del mismo tipo que el predictor 5 del codificador 1 es operable para realizar) para seleccionar uno de los conjuntos de coeficientes IIR a partir de la paleta de coeficientes IIR 8 predeterminada, y para configurar el filtro IIR 31 con el seleccionado de los conjuntos, y normalmente también para configurar el filtro FIR 33 en consecuencia. En algunas de estas implementaciones, el predictor 29 es operable para actualizar los filtros 31 y 33 de manera adaptativa. Las implementaciones alternativas descritas en este párrafo no serían adecuadas para reconstruir sin pérdidas datos codificados en un codificador sin pérdidas, a menos que puedan configurar los filtros 31 y 33 de manera que la configuración del predictor 29 coincida con la configuración de su contraparte en el codificador, para decodificar muestras codificadas con el predictor del codificador en dicha configuración.

En cualquier realización del decodificador de la invención que incluya tanto el filtro IIR 31 como el filtro FIR 33, cada vez que se determina la configuración de uno de los filtros IIR 31 y FIR 33 (o se actualiza), se determina la configuración de otro de los filtros 31 y 33 (o se actualiza). En casos típicos, esto se hace configurando ambos filtros 31 y 33 con coeficientes incluidos en un conjunto actual de datos de coeficientes de filtro (que han sido recibidos de un codificador y desempaquetados en la etapa 23). En estos casos, el codificador transmite al decodificador todos los coeficientes FIR e IIR requeridos de manera que el decodificador no tiene que realizar ningún cálculo y no necesita conocer la paleta IIR utilizada por el codificador (que se puede cambiar en cualquier momento sin necesidad de alterar los decodificadores existentes). En estos casos, la necesidad de transmisión de coeficientes (al decodificador desde el codificador) normalmente impone restricciones en el proceso de generación de la paleta de coeficientes IIR que se emplea en el codificador, ya que normalmente hay un número máximo de coeficientes IIR+FIR que se pueden enviar al decodificador, un número total máximo de etapas de filtro que se pueden utilizar (en el predictor del codificador y en el predictor del decodificador), y un número total máximo de bits que se pueden utilizar para los coeficientes transmitidos.

Con referencia de nuevo al decodificador 21 de la FIG. 3, los filtros 31 y 33 se implementan y configuran de manera que sus salidas combinadas, en respuesta a la secuencia de valores codificados V_x (generados en la etapa 27), son indicativas de un siguiente valor codificado previsto V_x en la secuencia. En la etapa 30, el predictor 29 resta cada valor actual de la salida del filtro 33 del valor actual de la salida del filtro 31 para generar una secuencia de valores previstos. En la etapa de cuantificación 32, el predictor 29 genera un secuencia de valores cuantificados realizando una operación de redondeo (p. ej., al entero más cercano) sobre cada valor previsto generado en la etapa 30.

En la etapa 34, el predictor 29 agrega cada valor actual cuantificado de la salida combinada de los filtros 31 y 33 (el próximo valor codificado previsto V_x que sale de la etapa 32) a cada valor actual de la secuencia de valores codificados V_x para generar una secuencia de valores codificados S_x .

Cada uno de los valores codificados S_x generados en la etapa 34 es una versión recuperada de manera exacta de una correspondiente de las muestras de audio codificadas S_x que fueron generadas en la etapa de re-matrización 3 del codificador 1 (y luego se sometieron a la codificación de predicción en la etapa del predictor 5 del codificador 1). Cada secuencia de valores cuantificados S_x generada en la etapa del predictor 29 es idéntica a una secuencia correspondiente de valores codificados S_x que fue generada en la etapa de re-matrización 3 del codificador 1.

Los valores cuantificados S_x generados en la etapa del predictor 29 se someten a la re-matrización en la etapa de re-matrización 41. En la etapa de re-matrización 41, la inversa de la codificación de re-matrización que fue realizada en

la etapa 3 del codificador 1 se realiza sobre los valores S_x , para recuperar las muestras de audio de entrada originales que fueron originalmente afirmadas en el codificador 1. Estas muestras recuperadas, etiquetadas como "muestras de audio de salida" en la FIG. 3, normalmente comprenden múltiples canales de muestras de audio.

5 Cada etapa de codificación del sistema de la FIG. 1 normalmente genera sus propios datos de cadena lateral. La etapa de re-matrización 3 genera coeficientes de re-matrización, el predictor 5 genera conjuntos actualizados de coeficientes de filtro IIR, el codificador de Huffman 13 genera un índice a una tabla de búsqueda de Huffman específica (para su uso por el decodificador 21, que debe implementar la misma tabla de búsqueda), y la etapa de representación de punto flotante de bloque 11 genera un exponente maestro para cada toque de muestras más mantisas de muestra individuales. La etapa de empaquetado 15 implementa una rutina de empaquetado maestra que toma todos los datos de cadena lateral de todas las etapas de codificación y los empaqueta todos juntos. La etapa de desempaqueado 23 en el decodificador de la FIG. 3 realiza la operación inversa (desempaquetado).

15 La etapa del predictor 29 del decodificador 21 aplica el mismo predictor implementado por el codificador 1 a una secuencia de valores ingresados al mismo (desde la etapa 27) para predecir un próximo valor en la secuencia. En una implementación típica de la etapa del predictor 29, cada valor previsto se agrega al valor correspondiente recibido de la etapa 27, para reconstruir una muestra codificada que fue salida de la etapa de re-matrización 3 del codificador 1. El decodificador 21 también realiza las inversas de las operaciones de codificación de Huffman y de re-matrización (realizadas en el codificador 1) para recuperar las muestras de entrada originales afirmadas en el codificador 1.

20 El sistema de la FIG. 1 se implementa preferiblemente como un codificador de audio digital sin pérdidas, y la salida decodificada (producida a la salida de una implementación compatible del decodificador de la FIG. 3) debe coincidir con la entrada al sistema de la FIG. 1 exactamente, bit por bit. Implementaciones preferidas del codificador y del decodificador (p. ej., el codificador de la FIG. 1 y el decodificador de la FIG. 3) comparten un protocolo común para expresar ciertas clases de señales de una forma más compacta, de manera que se reduce la velocidad de datos de los datos codificados que salen del codificador pero el decodificador puede recuperar la entrada de señal original al codificador.

25 El predictor 5 del sistema de la FIG. 1 utiliza una combinación de filtros IIR y FIR (filtro FIR 9 y filtro IIR 7). Trabajando juntos, los filtros generan una estimación de la próxima muestra de audio basándose en muestras previas. La estimación se resta (en la etapa 6) de la muestra actual, resultando en una muestra residual de amplitud reducida que se cuantifica y se afirma a la etapa 11 para codificación adicional. Una ventaja de utilizar un filtro de predicción que incluye tanto filtros de retroalimentación como de compensación (p. ej., el filtro IIR 7 y el filtro FIR 9) es que cada uno de los filtros de retroalimentación y compensación puede ser efectivo bajo condiciones de señal para las que se adapta mejor. Por ejemplo, el filtro FIR 9 puede compensar un pico en el espectro de la señal con menos coeficientes que el filtro IIR 7, mientras que lo contrario es cierto para una caída repentina en el espectro de la señal.

Alternativamente, algunos ejemplos útiles para entender la invención del filtro de predicción (y un codificador o decodificador en el que se implementa) incluyen solo un filtro de retroalimentación (IIR).

35 Para funcionar efectivamente, los coeficientes de los filtros FIR e IIR del predictor deben seleccionarse para coincidir con las características de la señal de entrada al predictor. Existen rutinas estándar eficientes para diseñar un filtro FIR dado un bloque de señal (p. ej., el método de recursión de Levinson-Durbin), pero no existen tales algoritmos para configurar un filtro IIR, ya sea de forma aislada o en colaboración con un filtro FIR. Para permitir la selección eficiente de coeficientes de filtro IIR (para configurar un filtro IIR de un predictor) de acuerdo con una clase de ejemplos útiles para entender la invención, se genera una paleta de conjuntos de coeficientes de filtro IIR pre-calculados que definen un conjunto de filtros IIR utilizando optimización no lineal restringida (p. ej., uno o ambos de un método Newtoniano restringido y de un método Simple restringido). Este proceso puede llevar mucho tiempo, ya que se realiza de forma preliminar a la configuración actual de un filtro de predicción que utiliza la paleta. La paleta que comprende los conjuntos de coeficientes de filtro IIR (cada conjunto definiendo un filtro IIR) se pone a disposición del sistema (p. ej., un codificador) que implementa el filtro de predicción a configurar. Normalmente, la paleta se almacena en el sistema (p. ej., el codificador) pero alternativamente puede almacenarse de forma externa al mismo y puede accederse cuando sea necesario. La memoria en la que se almacena la paleta es a veces referida en la presente memoria por conveniencia como la propia paleta (p. ej., la paleta 8 del predictor 5 es una memoria que almacena una paleta que ha sido generada). La paleta es preferiblemente lo suficientemente pequeña (suficientemente corta) para que el codificador puede intentar rápidamente cada filtro IIR determinado por un conjunto de coeficientes en la paleta, y elegir el que mejor funcione. Después de intentar cada filtro IIR candidato, un codificador (que implementa un filtro de predicción que incluye un filtro FIR así como el filtro IIR) puede realizar una eficiente recursión de Levinson-Durbin a la salida residual IIR (determinada utilizando el filtro IIR, configurado con el conjunto de coeficientes seleccionado) para determinar un conjunto óptimo de coeficientes de filtro FIR. El filtro FIR y el filtro IIR se configuran de acuerdo con la mejor combinación determinada de configuraciones IIR y FIR, y se utilizan para producir datos filtrados de predicción (p. ej., la secuencia de residuos que viaja desde la etapa de predicción 5 de la FIG. 1 hasta la etapa 11). En codificadores alternativos, los datos filtrados de predicción producidos por el filtro de predicción configurado (p. ej., los residuos producidos por la etapa 5 configurada en respuesta a cada bloque de entrada de muestras a la misma) se transmiten al decodificador sin codificarse más, junto con los coeficientes de filtro IIR seleccionados empleados para generar los datos (o con datos de coeficientes de filtro que identifican los coeficientes IIR seleccionados).

En un ejemplo útil para entender la invención, el codificador (p. ej., el codificador 1 de la FIG. 1) se implementa para operar con un tamaño de bloque de muestra que es variable en el siguiente sentido. Por ejemplo, como se ha señalado anteriormente en relación con la actualización adaptativa de las configuraciones de los filtros 7 y 9, el codificador 1 se opera preferiblemente para determinar cuántos micro-bloques de las muestras codificadas (generadas en la etapa 3) para codificar aún más utilizando cada configuración determinada de los filtros 7 y 9. En dichos ejemplos preferidos útiles para entender la invención, el codificador 1 determina efectivamente el tamaño de un "macro-bloque" de las muestras codificadas (generadas en la etapa 3) que se codificará utilizando cada configuración determinada de los filtros 7 y 9 (sin actualizar la configuración). Por ejemplo, un ejemplo preferido útil para entender la invención del predictor 5 del codificador 1 puede determinar el tamaño de cada macro-bloque de las muestras codificadas (generadas en la etapa 3) a codificar, utilizando cada configuración determinada de los filtros 7 y 9, como un número N (donde N está en el rango de $1 \leq N \leq 128$) de los micro-bloques. Para determinar el número N óptimo, el predictor 5 puede operar para actualizar los filtros 7 y 9 una vez por cada micro-bloque (p. ej., que consta de 48 muestras) de muestras y para filtrar cada uno de una secuencia de micro-bloques, luego actualizar los filtros 7 y 9 (p. ej., en cualquiera de las formas descritas en la presente memoria) una vez por cada secuencia de X micro-bloques y filtrar cada uno de una secuencia de dichos grupos de micro-bloques, y luego actualizar los filtros 7 y 9 una vez por cada grupo más grande de micro-bloques y filtrar cada uno de una secuencia de dichos grupos más grandes de micro-bloques, y así sucesivamente en una secuencia (p. ej., hasta un grupo de 128 de los micro-bloques), y determinar a partir de los datos resultantes el tamaño de macro-bloque óptimo (el número N óptimo de los micro-bloques por macro-bloque). Por ejemplo, el tamaño de macro-bloque óptimo puede ser el número máximo de micro-bloques que pueden agruparse juntos para hacer cada macro-bloque sin incrementar de manera inaceptable el nivel RMS de los residuos generados por el predictor 5 (o el nivel RMS del flujo de datos de salida generado por el codificador 1, incluyendo todos los datos de sobrecarga).

En algunos ejemplos útiles para entender la invención, la actualización adaptativa del filtro IIR 7 y del filtro FIR 9 se realiza una vez (o Z veces, donde Z es algún número determinado) por macro-bloque (p.ej., una vez por cada 128 micro-bloques de muestras a codificar por el codificador 1), pero no más de una vez por micro-bloque de muestras a codificar por el codificador 1.

En algunos ejemplos útiles para entender la invención, la operación de codificación del codificador 1 está deshabilitada para las primeras X (p. ej., $X = 8$) muestras en cada macro-bloque (el filtro IIR 7 y el filtro FIR 9 pueden actualizarse durante los periodos en los que la operación de codificación está deshabilitada). Las X muestras no codificadas por macro-bloque se pasan al decodificador.

Algunos ejemplos útiles para entender la invención del codificador 1 restringen los intervalos entre eventos de actualización adaptativa de las configuraciones del filtro de predicción (p. ej., la frecuencia máxima a la que se permite la actualización de los filtros 7 y 9), p. ej., para optimizar la eficiencia de la codificación. Cada vez que se reconfigura el filtro IIR 7 en el codificador 1 (implementado como un codificador sin pérdidas), hay un cambio de estado en el codificador que requiere que datos de sobrecarga (datos de cadena lateral) indicativos del nuevo estado se transmitan para permitir al decodificador 21 tener en cuenta cada cambio de estado durante la decodificación. Sin embargo, si el cambio de estado del codificador sucede por alguna razón que no es una reconfiguración del filtro IIR (p. ej., un cambio de estado que se produce al inicio del procesamiento de un nuevo macro-bloque de muestras), los datos de sobrecarga indicativos del nuevo estado deben también transmitirse al decodificador 21 de manera que la reconfiguración de los filtros 7 y 9 puede realizarse en este momento sin agregar (o sin agregar de manera significativa o intolerable) a la cantidad de datos de sobrecarga que deben transmitirse. Así, algunos ejemplos útiles para entender la invención del codificador 1 se configuran para realizar una operación de determinación de continuidad para determinar cuando hay un cambio de estado del codificador, y para controlar la sincronización de las operaciones para reconfigurar los filtros 7 y 9 en consecuencia (p. ej., de manera que la reconfiguración de los filtros 7 y 9 se aplase hasta que ocurra un evento de cambio de estado al inicio de un nuevo macro-bloque).

A continuación describimos cuatro aspectos. Los dos primeros son métodos preferidos (y los sistemas programados para realizarlos) para generar una paleta de coeficientes de filtro IIR para proporcionar a un codificador, para su uso en la configuración de un filtro de predicción del codificador (donde el filtro de predicción incluye un filtro IIR y también opcionalmente un filtro FIR). Los dos segundos son métodos preferidos (y los sistemas programados para realizarlos) para utilizar la paleta para configurar un filtro de predicción de un codificador, donde el filtro de predicción incluye un filtro IIR y también opcionalmente un filtro FIR.

Normalmente, un procesador (programado apropiadamente con firmware o software) se opera para generar una paleta maestra de coeficientes de filtro IIR para proporcionar a un codificador. Como se describió anteriormente, cada conjunto de coeficientes en la paleta maestra puede generarse realizando optimización no lineal sobre un conjunto (un "conjunto de entrenamiento") de señales de entrada (p. ej., muestras de datos de audio), sujeta a al menos a una restricción. Ya que este proceso puede producir una paleta maestra inaceptablemente grande, se puede realizar un proceso de poda sobre la paleta maestra (para eliminar los conjuntos de coeficientes IIR de los mismos y generar así una paleta final más pequeña de conjuntos de coeficientes IIR) basándose en alguna combinación de la acumulación del histograma y de la mejora neta proporcionada por cada filtro IIR candidato sobre el conjunto de entrenamiento.

En un ejemplo útil para entender la invención, una paleta maestra de coeficientes IIR se recorta como sigue para obtener una paleta final. Para cada bloque de muestras de señal de cada señal en un (posiblemente diferente) conjunto

de entrenamiento de señales (posiblemente diferente del conjunto de entrenamiento utilizado para generar la paleta maestra), para cada filtro IIR candidato en la paleta maestra, se calcula un filtro FIR correspondiente utilizando recursión de Levinson-Durbin. Los residuos generados por el filtro IIR y el filtro FIR candidatos combinados se evalúan, y los coeficientes IIR que determinan el filtro IIR de la combinación del filtro IIR y del filtro FIR que produce la señal residual que tiene un menor nivel RMS se seleccionan para su inclusión en la paleta final (la selección puede estar condicionada por la Q máxima y la precisión deseada de la combinación de filtros IIR/FIR). Se pueden acumular histogramas de uso total de cada filtro y de mejora neta. Después de procesar el conjunto de entrenamiento, los filtros menos efectivos se recortan de la paleta. El procedimiento de entrenamiento puede repetirse hasta que se alcance una paleta del tamaño deseado.

En un ejemplo útil para entender la invención, el método genera la paleta de coeficientes de filtro IIR de manera que cada filtro IIR determinado por cada conjunto de coeficientes en la paleta tenga un orden que puede seleccionarse a partir de un número de diferentes órdenes posibles. Por ejemplo, se considera uno de los conjuntos (un "primer conjunto") de coeficientes IIR en dicha paleta. El primer conjunto puede ser útil para configurar un filtro IIR que tiene un orden que se puede seleccionar en el siguiente sentido: un primer subconjunto (de los coeficientes en el primer conjunto) determina una implementación de primer orden seleccionada del filtro IIR, y al menos otro subconjunto (de los coeficientes en el primer conjunto) determina una implementación de N-ésimo orden seleccionada del filtro IIR (donde N es un entero mayor que 1, p. ej., N = 4 para implementar un filtro de cuarto orden). En una realización preferida, el filtro de predicción a configurar utilizando la paleta (p. ej., una implementación preferida del filtro de predicción implementado por la etapa 5 del codificador 1) incluye un filtro IIR y un filtro FIR, y durante la configuración del filtro de predicción utilizando la paleta, los órdenes de estos filtros se pueden seleccionar sujetos a las restricciones de que el orden del filtro IIR está en el rango de 0 a X incluido (p. ej., X = 4), el orden del filtro FIR está en el rango de 0 a Y (p. ej., Y = 12), y los órdenes seleccionados del filtro IIR y del filtro FIR pueden sumar hasta un máximo de Z (p. ej., Z = 12).

Como se ha señalado, cada conjunto de coeficientes en la paleta puede generarse realizando optimización no lineal sobre un conjunto (un "conjunto de entrenamiento") de señales de entrada (p. ej., muestras de datos de audio), sujeta a al menos a una restricción. En algunos ejemplos útiles para entender la invención, esto se hace como sigue (asumiendo que el filtro de predicción a configurar utilizando la paleta aplicará tanto a un filtro FIR como a un filtro IIR para generar residuos). Para cada conjunto de prueba de coeficientes IIR de cada recursión del optimizador sobre cada bloque de muestra, se realiza una rutina de diseño FIR de Levinson-Durbin para obtener coeficientes del filtro de predicción FIR óptimos que corresponden al filtro de predicción IIR determinado por el conjunto de prueba. Se determina una mejor combinación de orden del filtro IIR/FIR y de valores de coeficientes IIR (y FIR correspondientes) basándose en el residuo de predicción mínimo, condicionado por las limitaciones en la sobrecarga de transmisión, Q de filtro máxima, precisión de coeficientes numéricos, y estabilidad. Para cada señal en el conjunto de prueba, el conjunto de coeficientes IIR de prueba incluido en una "mejor" combinación IIR/FIR determinada por la optimización se incluye en la paleta (si no está ya presente). El proceso continua hasta acumular un conjunto de coeficientes IIR en la paleta maestra para cada señal en todo el conjunto de entrenamiento.

Un método preferido (y el sistema programado para realizarlo) para utilizar una paleta de coeficientes IIR determinada para configurar un filtro de predicción de un codificador (donde el filtro de predicción incluye un filtro IIR y un filtro FIR), incluye los siguientes pasos: para cada bloque de un conjunto de datos de entrada, se aplica cada filtro IIR determinado por los conjuntos de coeficientes en la paleta para generar los primeros residuos, se determina una mejor configuración de filtro FIR para cada filtro IIR aplicando un método de recursión de Levinson-Durbin a los primeros residuos (p. ej., para determinar una configuración FIR) que, cuando se aplica a los primeros residuos, resulta en un conjunto de residuos de predicción que tiene nivel más bajo (p. ej., nivel RMS más bajo) que incluye el tener en cuenta la sobrecarga de transmisión de coeficientes (p. ej., que incluye la sobrecarga requerida para transmitir con cada conjunto de residuos de predicción y que elige la configuración FIR que minimiza el nivel de los residuos de predicción incluyendo la sobrecarga), y la configuración del filtro de predicción con la mejor combinación determinada de coeficientes IIR y de coeficientes FIR.

Un método preferido (y el sistema programado para realizarlo) para utilizar una paleta de coeficientes IIR determinada para configurar un filtro de predicción de un codificador (donde el filtro de predicción incluye un filtro IIR y un filtro FIR), incluye los siguientes pasos: utilizar la paleta para determinar una mejor combinación de coeficientes IIR y de coeficientes FIR, y establecer el estado del filtro de predicción utilizando la mejor combinación determinada de coeficientes IIR y de coeficientes FIR de una manera que tiene en cuenta (y preferiblemente para maximizar) la continuidad de la señal de salida (p. ej., utilizando la optimización de mínimos cuadrados). Por ejemplo, el filtro de predicción puede no reconfigurarse con el conjunto recientemente determinado de coeficientes IIR y FIR si hacerlo requeriría la transmisión de datos de sobrecarga inaceptables (p. ej., para indicar un cambio de estado resultante de la reconfiguración del decodificador), o el filtro de predicción puede reconfigurarse con el conjunto recientemente determinado de coeficientes IIR y FIR en un momento que coincida con un cambio de estado al inicio de un nuevo macro-bloque de muestras para codificar por predicción.

Para permitir el uso práctico de un predictor de retroalimentación (un predictor que incluye un filtro de predicción que incluye un filtro de retroalimentación, con o sin aumento por predicción de compensación), un codificador que incluye el predictor se proporciona con una lista ("paleta") de coeficientes de filtro de retroalimentación pre-calculados. Cuando

hay que seleccionar un nuevo filtro, el codificador necesita intentar solo cada filtro de retroalimentación (IIR) determinado por la paleta (sobre un conjunto de valores de datos de entrada, p. ej., un bloque de muestras de datos de audio) para determinar la mejor opción, lo cual es generalmente un cálculo rápido si la paleta no es demasiado grande. Por ejemplo, un mejor conjunto de coeficientes para el predictor puede determinarse intentando cada conjunto de coeficientes en la paleta, y seleccionando el conjunto de coeficientes que resulta en una señal residual que tiene un menor nivel RMS como el “mejor” conjunto de coeficientes (donde se genera una señal residual para cada conjunto de coeficientes aplicando el filtro de predicción, configurado con dicho conjunto, a una señal de entrada, p. ej., a la señal de entrada a codificar u otra señal que tiene características similares a la señal de entrada a codificar). Normalmente, es mejor minimizar el nivel RMS del residuo, ya que esto permitirá un procesador de punto flotante de bloque (u otra etapa de codificación) para minimizar los bits de los datos codificados generados así.

En algunos ejemplos útiles para entender la invención, el método para seleccionar una mejor combinación de las configuraciones de filtro FIR/IIR (o una mejor configuración de filtro IIR) para un codificador de predicción en un codificador de múltiples etapas, donde el codificador de múltiples etapas incluye otras etapas de codificación (p. ej., etapas de codificación de punto flotante de bloque y de Huffman) así como el codificador de predicción, considera que el resultado de aplicar todas las etapas de codificación (incluyendo el predictor) a una señal de entrada (con el codificador de predicción configurado con cada conjunto candidato de coeficientes IIR determinado por una paleta). La combinación seleccionada de coeficientes de filtro FIR/IIR (o el mejor conjunto de coeficientes IIR) puede ser una que resulte en la velocidad de datos neta más baja de la salida completamente codificada del codificador de múltiples etapas. Sin embargo, dado que dicho cálculo puede llevar mucho tiempo, el nivel RMS (teniendo en cuenta también la sobrecarga de cadena lateral) de la salida de la etapa de codificación de predicción solo puede utilizar el criterio para determinar una mejor combinación de coeficientes de filtro FIR/IIR (o un mejor conjunto de coeficientes IIR) para la etapa del codificador de predicción de dicho codificador de múltiples etapas.

También, ya que una reconfiguración de un filtro de predicción en un codificador (para implementar un nuevo conjunto de coeficientes de filtro IIR, o de coeficientes de filtro IIR y FIR), puede introducir un breve transitorio que aumentará la velocidad de datos de la salida del codificador, a veces es preferible tener en cuenta la sobrecarga asociada con cada uno de dichos transitorios para determinar la sincronización de una reconfiguración contemplada del filtro de predicción.

Como se ha señalado anteriormente, se utiliza un método de recursión (p. ej., una recursión de Levinson-Durbin) para determinar un conjunto de coeficientes de filtro FIR para configurar el filtro FIR de un filtro de predicción, donde el filtro de predicción incluye tanto un filtro FIR como un filtro IIR, y un conjunto de coeficientes de filtro IIR (para configurar el filtro IIR) ya ha sido determinado (p. ej., utilizando cualquier realización del método de la invención). En este contexto, el filtro FIR puede ser un filtro predictor de compensación de N-ésimo orden, y el método de recursión puede tomar como entrada un bloque de muestras (p. ej., muestras generadas aplicando el filtro IIR, configurado con el conjunto determinado de coeficientes de filtro IIR, a los datos), y determinar utilizando cálculos recursivos un conjunto óptimo de coeficientes de filtro FIR para el filtro FIR. Los coeficientes pueden ser óptimos en el sentido de que minimizan el error cuadrático medio de una señal residual. Cada iteración durante la recursión (antes de converger para determinar un conjunto óptimo de coeficientes de filtro FIR) normalmente asume un conjunto diferente de coeficientes de filtro FIR (a veces referido en la presente memoria como un “conjunto candidato” de coeficientes de filtro FIR). En algunos casos, la recursión puede comenzar por encontrar los coeficientes del predictor de 1^{er} orden óptimos, luego utilizar estos para encontrar los coeficientes del predictor de 2^o orden óptimos, luego utilizar estos para encontrar los coeficientes del predictor de 3^{er} orden óptimos, y así hasta que un conjunto óptimo de coeficientes de filtro para el filtro predictor de compensación de N-ésimo orden haya sido determinado.

En realizaciones típicas, el sistema de la invención incluye un procesador de propósito general o especial programado con software (o firmware) y/o configurado de otra manera para realizar una realización del método de la invención. Un procesador de señal digital (DSP) adecuado para procesar los datos de entrada esperados será una implementación preferida para muchas aplicaciones. En algunas realizaciones, el sistema de la invención es un procesador de propósito general, acoplado a los datos de entrada recibidos, y programado (con software apropiado) para generar datos de salida en respuesta a los datos de entrada realizando una realización del método de la invención. En algunas realizaciones, el sistema de la invención es un decodificador (implementado como un DSP), u otro DSP, que se programa y/o configura de otra manera para realizar una realización del método de la invención sobre los datos.

La FIG. 4 es una vista en alzado de un disco óptico legible por ordenador 50, en el que se almacena código para implementar el método (p. ej., para generar una paleta de coeficientes de filtro IIR, y/o para realizar una operación de filtrado de predicción sobre las muestras de datos y actualizar de manera adaptativa la configuración de un filtro IIR y de un filtro FIR del filtro de predicción empleado para realizar el filtrado). Por ejemplo, el código puede ser ejecutado por un procesador para generar una paleta de coeficientes de filtro IIR (p. ej., paleta 8). O, el código puede cargarse en una realización del codificador 1 para programar el codificador 1 para realizar una operación filtrado de predicción (en el predictor 5) sobre las muestras de datos y para actualizar de manera adaptativa la configuración del filtro IIR 7 y del filtro FIR 9, o en el decodificador 21 para programar el decodificador 21 para realizar una operación de filtrado de predicción (en el predictor 29) sobre las muestras de datos y para actualizar de manera adaptativa la configuración del filtro IIR 31 y del filtro FIR 33.

5 Aunque se han descrito en la presente memoria realizaciones específicas de la presente invención y aplicaciones de la invención, será evidente para los expertos en la técnica que son posibles múltiples variaciones en las realizaciones y aplicaciones descritas en la presente memoria sin salirse del alcance de la invención reivindicada en la presente memoria. Debe entenderse que si bien se han mostrado y descrito ciertas formas de la invención, la invención no debe limitarse a las realizaciones específicas descritas y mostradas o a los métodos específicos descritos.

REIVINDICACIONES

1. Un decodificador acoplado a los datos codificados recibidos, dicho decodificador incluye:

5 una etapa de desempaquetado (23) configurada para, en respuesta a los datos codificados, desempaquetar los valores codificados de Huffman, dichas palabras de cadena, datos de coeficientes de filtro indicativos de los conjuntos de coeficientes primero y segundo, y valores que determinan el tamaño de un macro-bloque de valores codificados de Huffman, siendo los valores utilizados para determinar los intervalos en los que un primer y un segundo filtro del decodificador deben reconfigurarse con el primer y el segundo conjunto de coeficientes, respectivamente,

una etapa de decodificación de Huffman (25) configurada para decodificar los valores codificados de Huffman;

10 una etapa de decodificación de representación de punto flotante de bloque (27) configurada para generar datos decodificados parcialmente en respuesta a los valores decodificados de Huffman, datos decodificados parcialmente que comprenden una pluralidad de valores, cada valor correspondiente a una suma de un residuo cuantificado de una muestra de los datos codificados y de los bits más significativos de la muestra correspondientes al residuo cuantificado;

15 un predictor (29), acoplado a la etapa de decodificación de representación de punto flotante de bloque (27) y que incluye el primer filtro, que es un filtro de respuesta de impulso finito (FIR) conectado en una configuración de retroalimentación, y el segundo filtro, que es un filtro FIR conectado en una configuración de compensación, y configurado para generar datos filtrados de predicción en respuesta a los datos decodificados parcialmente, en donde el primer filtro se configura con el primer conjunto de coeficientes y el segundo filtro se configura con el segundo conjunto de coeficientes en los intervalos determinados por la etapa de desempaquetado; y

20 una etapa de re-matrización (41) configurada para recuperar muestras de audio de salida a partir de los datos filtrados de la predicción.

2. El decodificador de la reivindicación 1, en donde los datos de los coeficientes de filtro son los conjuntos de coeficientes primero y segundo.

3. El decodificador de cualquiera de las reivindicaciones 1-2, en donde el decodificador es un aparato de decodificación sin pérdidas.

25

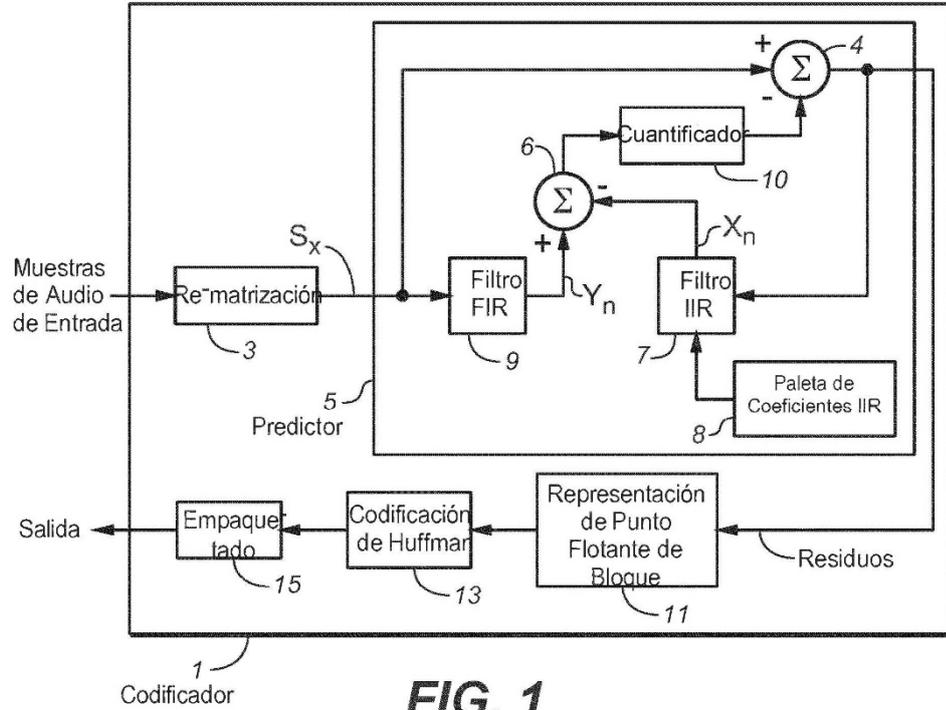


FIG. 1

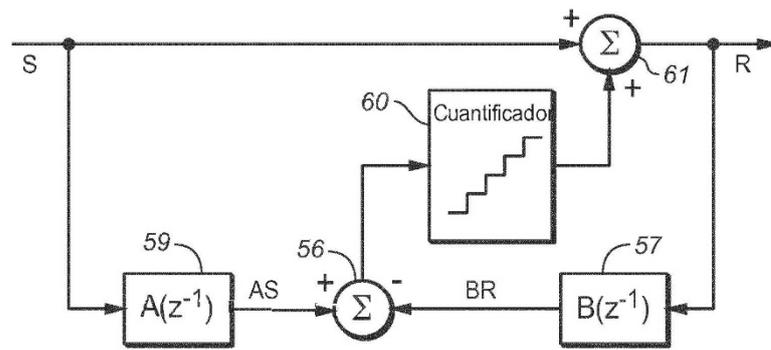


FIG. 2 (TÉCNICA ANTERIOR)

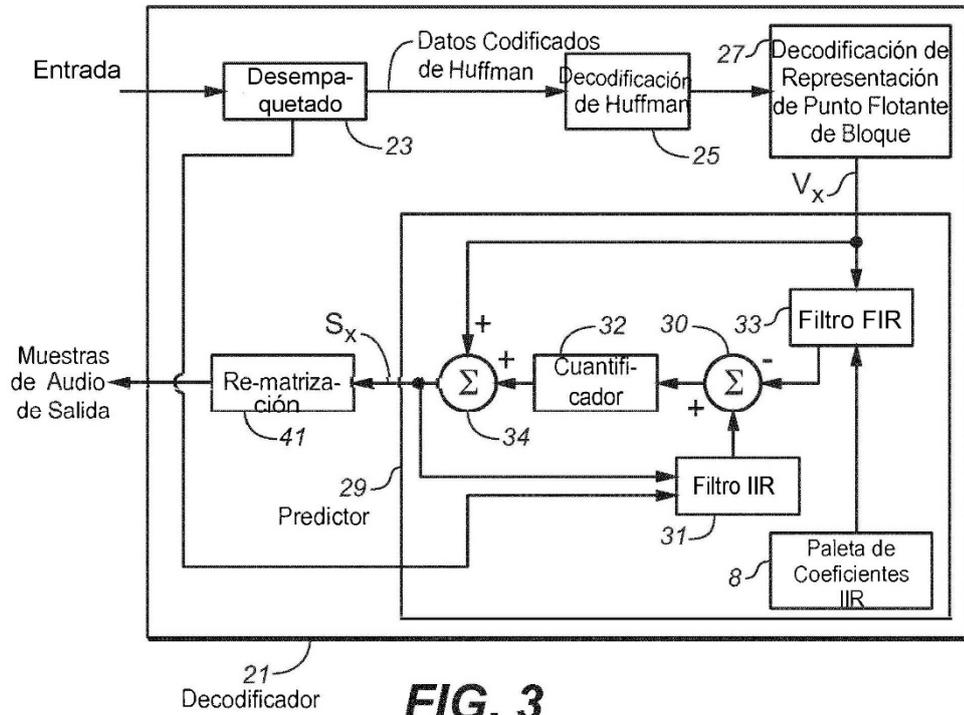


FIG. 3

