



19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA

11 Número de publicación: **2 358 213**

51 Int. Cl.:
G10L 19/00 (2006.01)
G10L 19/08 (2006.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

96 Número de solicitud europea: **06749340 .3**
96 Fecha de presentación : **05.04.2006**
97 Número de publicación de la solicitud: **1886306**
97 Fecha de publicación de la solicitud: **13.02.2008**

54 Título: **Flujo redundante de bits de audio y métodos de procesamiento de flujo de bits de audio.**

30 Prioridad: **31.05.2005 US 142605**

45 Fecha de publicación de la mención BOPI:
06.05.2011

45 Fecha de la publicación del folleto de la patente:
06.05.2011

73 Titular/es: **MICROSOFT CORPORATION**
One Microsoft Way
Redmond, Washington 98052-6399, US

72 Inventor/es: **Wang, Tian;**
Koishida, Kazuhito;
Khalil, Hosam A.;
Sun, Xiaoqin y
Chen, Wei-Ge

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

ES 2 358 213 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

Flujo redundante de bits de audio y métodos de procesamiento de flujo de bits de audio

Las técnicas y herramientas descritas se refieren a códecs (codificadores-descodificadores), y a codificación sub-banda, tablas de código, y/o codificación redundante.

5 ANTECEDENTES

Con el surgimiento de las redes de telefonía inalámbrica digital, las emisiones de audio en tiempo real sobre la red Internet, y la telefonía de Internet, el procesamiento digital y la distribución de la voz se han hecho habituales. Los ingenieros utilizan una variedad de técnicas para procesar eficientemente la voz manteniendo al mismo tiempo la calidad. Para comprender estas técnicas, resulta de ayuda comprender cómo la información de audio es representada y procesada en un ordenador.

I. Representación de la Información de Audio en un Ordenador

Un ordenador procesa información de audio como una serie de números que representan el audio. Un solo número puede representar una muestra de audio, que es un valor de amplitud en un momento concreto. Diversos factores afectan a la calidad del audio, incluyendo la profundidad de la muestra y la velocidad de muestreo.

15 La profundidad de la muestra (o precisión) indica el rango de números utilizados para representar una muestra. Más valores posibles para cada muestra producen habitualmente una salida de calidad superior, debido a que pueden ser representadas variaciones más sutiles en la amplitud. Una muestra de ocho bits tiene 256 posibles valores, mientras que una muestra de 16 bits tiene 65 536 posibles valores.

20 La velocidad de muestreo (medida habitualmente como el número de muestras por segundo) afecta asimismo a la calidad. Cuanto mayor es la velocidad de muestreo, mayor es la calidad debido a que pueden ser representadas más frecuencias de sonido. Algunas velocidades de muestreo habituales son 8000, 11 025, 22 050, 32 000, 44 100, 48 000 y 96 000 muestras/segundo (Hz). La tabla 1 muestra varios formatos de audio con diferentes niveles de calidad, junto con los correspondientes costes de velocidad binaria en bruto.

Tabla 1: Velocidades binarias para audio de diferentes calidades

25

Profundidad de la Muestra (bits/muestra)	Velocidad de Muestreo (muestras/segundo)	Modo del Canal	Velocidad binaria en Bruto (bits/segundo)
8	8 000	mono	64 000
8	11 025	mono	88 200
16	44 100	estéreo	1 411 200

Tal como muestra la tabla 1, el coste del audio de alta calidad es una velocidad binaria elevada. La información de audio de alta calidad consume grandes cantidades de almacenamiento informático y de capacidad de transmisión. Muchos ordenadores y redes informáticas carecen de los recursos para procesar el audio digital en bruto. La compresión (denominada asimismo codificación o cifrado) reduce el coste de almacenar y transmitir información de audio, convirtiendo la información en una versión con velocidad binaria menor. La compresión puede ser sin pérdidas (en la cual no sufre la calidad) o con pérdidas (en la cual sufre la calidad, pero la reducción de la velocidad binaria a partir de la compresión sin pérdidas subsiguiente, es más dramática). La descompresión (denominada asimismo descodificación) extrae una versión reconstruida de la información original, a partir de la forma comprimida. Un códec es un sistema de codificador/descodificador.

II. Codificadores y Descodificadores de Voz

Un objetivo de la compresión de audio es representar digitalmente señales de audio para proporcionar una calidad de señal máxima para una cantidad de bits dada. Dicho de otra forma, este objetivo consiste en representar las

señales de audio con los mínimos bits para un nivel de calidad dado. Otros objetivos tales como la resistencia a errores de transmisión y la limitación del retardo global debido a la codificación/transmisión/descodificación, son de aplicación en algunos escenarios.

5 Diferentes clases de señales de audio tienen características diferentes. La música se caracteriza por rangos grandes de frecuencias y amplitudes, y a menudo incluye dos o más canales. Por otra parte, la voz se caracteriza por rangos menores de frecuencias y amplitudes, y habitualmente se representa en un solo canal. Ciertos códecs y técnicas de procesamiento están adaptados para música y audio en general; otros códecs y técnicas de procesamiento están adaptados para voz.

10 Un tipo de códec de voz convencional utiliza predicción lineal para conseguir compresión. La codificación de la voz comprende varias etapas. El codificador halla y cuantifica coeficientes para un filtro de predicción lineal, que se utiliza para predecir valores de muestra como combinaciones lineales de valores de muestra precedentes. Una señal residual (representada como una señal de "excitación") indica partes de la señal original no predichas con precisión mediante el filtrado. En algunas etapas, el códec de voz utiliza diferentes técnicas de compresión para segmentos con voz (caracterizados por la vibración de las cuerdas vocales), segmentos sin voz, y segmentos de silencio, 15 puesto que las diferentes clases de discurso hablado tienen características diferentes. Habitualmente, los segmentos con voz presentan patrones de expresión de la voz muy repetitivos, incluso en el dominio residual. Para segmentos con voz, el codificador consigue más compresión comparando la señal residual actual con ciclos residuales previos, y codificando la señal residual actual en términos de información de retardo o desfase relativa a los ciclos previos. El codificador trata otras discrepancias entre la señal original y la representación codificada, predicha, utilizando tablas de código diseñadas especialmente. 20

Muchos códecs de voz explotan de algún modo la redundancia temporal en una señal. Tal como se ha mencionado anteriormente, un método común utiliza predicción a largo plazo de parámetros de paso para predecir una señal de excitación actual en términos de retardo o desfase relativos a ciclos de excitación previos. Explotar la redundancia temporal puede mejorar sensiblemente la eficiencia de la compresión en términos de calidad y velocidad binaria, 25 pero al coste de introducir en el códec una dependencia con la memoria (un descodificador depende de una parte, descodificada previamente, de la señal para descodificar correctamente otra parte de la señal). Muchos códecs de voz eficientes tienen una dependencia significativa con la memoria.

Aunque los códecs de voz como los descritos anteriormente tienen un buen rendimiento global para muchas aplicaciones, tienen varios inconvenientes. En concreto, aparecen varios inconvenientes cuando los códecs de voz 30 son utilizados junto con recursos de red dinámicos. En dichos escenarios, la voz codificada puede perderse debido a una escasez temporal del ancho de banda, o a otros problemas.

A. Códecs de Banda Estrecha y de Banda Ancha

Muchos códecs de voz estándar fueron diseñados para señales de banda estrecha con una velocidad de muestreo de ocho kHz. Si bien la velocidad de muestreo de ocho kHz es adecuada en muchas situaciones, en otras 35 situaciones pueden ser deseables velocidades de muestreo superiores, tal como para representar frecuencias superiores.

Habitualmente, las señales de voz con velocidades de muestreo de, por lo menos, 16 kHz se denominan voz de banda ancha. Si bien estos códecs de banda ancha son deseables para representar patrones de voz de alta frecuencia, habitualmente requieren velocidades binarias superiores que los códecs de banda estrecha. Dichas 40 velocidades binarias pueden no ser factibles en algunos tipos de redes o bajo ciertas condiciones de la red.

B. Dependencia Ineficiente de la Memoria en Condiciones de Red Dinámicas

Cuando la voz codificada está fallando, tal como por haberse perdido, retrasado, deteriorado o hacerse inutilizable por otra razón en el tránsito o en cualquier otro lugar, el rendimiento de los códecs de voz puede verse afectado debido a la dependencia de la memoria con la información perdida. La pérdida de información por una señal de 45 excitación dificulta la reconstrucción posterior que depende de la señal perdida. Si se han perdido ciclos anteriores, la información de desfase puede no ser útil, puesto que apunta a información que el descodificador no tiene. Otro ejemplo de dependencia con la memoria es la interpolación de coeficientes de filtro (utilizada para suavizar las transiciones entre diferentes filtros de síntesis, especialmente para señales con voz). Si se pierdan los coeficientes de filtro para una trama, los coeficientes de filtro para las tramas subsiguientes pueden tener valores incorrectos.

50 Los descodificadores utilizan diversas técnicas para ocultar errores debidos a pérdidas de paquetes y a otras pérdidas de información, pero estas técnicas de ocultación rara vez ocultan los errores por completo. Por ejemplo, el descodificador repite parámetros anteriores o parámetros estimados, basándose en información de descodificada

correctamente. Sin embargo, la información de desfase puede ser muy sensible, y las técnicas anteriores no son especialmente eficaces para la ocultación.

5 En la mayor parte de los casos, los descodificadores se recuperan finalmente de los errores debidos a información perdida. Cuando son recibidos y descodificados los paquetes, los parámetros son ajustados gradualmente hacia sus valores correctos. Sin embargo, es probable que se degrade la calidad hasta que el descodificador pueda recuperar el estado interno correcto. En muchos de los códecs de voz más eficientes, la calidad de la reproducción se degrada durante un periodo de tiempo prolongado (por ejemplo, hasta un segundo), provocando una gran distorsión y, a menudo, haciendo la voz ininteligible. Los tiempos de recuperación son más rápidos cuando se produce un cambio significativo, tal como una trama de silencio, puesto que esto proporciona un punto de reajuste natural para muchos parámetros. Algunos códecs son más robustos frente a pérdidas de paquetes debido a que eliminan sus dependencias entre tramas. Sin embargo, dichos códecs requieren velocidades binarias significativamente superiores para conseguir la misma calidad de voz que el códec CELP tradicional con dependencias entre tramas.

Para tratar con las dependencias entre tramas, en la técnica anterior han sido propuestas varias soluciones.

15 El documento US 6 757 654 da a conocer datos redundantes incrustados en una trama, acerca de la trama anterior. Cuando se pierde una trama actual, los datos redundantes de la siguiente trama válida son utilizados para reconstruir dicha trama actual. Esta solución no es utilizable en el caso de una trama anterior perdida, para descodificar una trama actual con dependencia a largo plazo sobre dicha trama anterior.

20 Otro documento, "A study of design compromises for speech coders in packet networks" ("un estudio de compromisos de diseño para codificadores de voz en redes por paquetes"), de R. Lefebvre y otros, PROCEEDINGS IEEE ICASSP' 04, 17 a 21 de mayo de 2004, páginas 265 a 268, da a conocer una solución que consiste en agrupar para un paquete dado, datos en torno a parámetros ITU-T G.729 de tramas actuales y tramas anteriores. Sin embargo, dicho documento no da a conocer cómo proteger el contenido de la propia tabla de código adaptativo.

25 Dada la importancia de la compresión y la descompresión para representar señales de voz en sistemas informáticos, no sorprende que la compresión y la descompresión de voz hayan atraído recursos y actividad de estandarización. Sin embargo, cualesquiera que sean ventajas de las técnicas y herramientas anteriores, no poseen las ventajas de las técnicas y herramientas descritas en el presente documento.

RESUMEN

Resumiendo, la invención se define mediante un flujo de bits acorde con la reivindicación 1, y métodos para el procesamiento de un flujo de bits acordes con las reivindicaciones 5 y 10.

BREVE DESCRIPCIÓN DE LOS DIBUJOS

La figura 1 es un diagrama de bloques de un entorno informático adecuado, en el que pueden ser implementadas una o más de las realizaciones descritas.

La figura 2 es un diagrama de bloques de un entorno de red con el cual pueden ser implementadas una o más de las realizaciones descritas.

35 La figura 3 es un gráfico que representa un conjunto de respuestas de frecuencia para una estructura de sub-banda que puede ser utilizada para codificación de sub-banda.

La figura 4 es un diagrama de bloques de un codificador de banda de voz en tiempo real, junto con el cual pueden ser implementadas una o más de las realizaciones descritas.

40 La figura 5 es un diagrama de bloques que representa la determinación de parámetros de la tabla de código, en una implementación.

La figura 6 es un diagrama de bloques de un descodificador de banda de voz en tiempo real, junto con el cual pueden ser implementadas una o más de las realizaciones descritas.

La figura 7 es un diagrama de un histórico de señales de excitación, que incluye una trama actual y una parte re-codificada de una trama anterior.

45 La figura 8 es un diagrama de flujo que describe la determinación de parámetros de tabla de código para una etapa de tabla de código aleatoria extra, en una implementación.

La figura 9 es un diagrama de bloques de un descodificador de banda de voz en tiempo real, que utiliza una etapa de tabla de código aleatoria extra.

La figura 10 es un diagrama de formatos de flujo de bits para tramas que comprenden información para diferentes técnicas de codificación redundante, que pueden ser utilizadas con algunas implementaciones.

5 La figura 11 es un diagrama de formatos de flujos de bits para paquetes que comprenden tramas con información de codificación redundante, que pueden utilizarse con algunas implementaciones.

DESCRIPCIÓN DETALLADA

10 Las realizaciones descritas están dirigidas a técnicas y herramientas para el procesamiento de información de audio, en la codificación y la descodificación. Con estas técnicas se mejora la calidad de voz extraída de un códec de voz, tal como un códec de voz en tiempo real. Dichas mejoras pueden resultar de la utilización de varias técnicas y herramientas por separado, o en combinación.

Dichas técnicas y herramientas pueden comprender la codificación y/o descodificación de sub-bandas utilizando técnicas de predicción lineal, tales como CELP.

15 Las técnicas pueden incluir asimismo tener múltiples etapas de tablas de código fijas, incluyendo tablas de código fijas de impulsos y/o aleatorias. El número de etapas de tabla de código puede variarse para maximizar la calidad para una velocidad de bits data. Adicionalmente, una tabla de código adaptativo puede ser activada o desactivada, dependiendo de factores tales como la velocidad binaria deseada y las características de la trama o sub-trama actual.

20 Además, las tramas pueden comprender información codificada redundante para parte o la totalidad de una trama previa, de la cual depende la trama actual. Esta información puede ser utilizada por el descodificador para descodificar la trama actual si se ha perdido la trama previa, sin que se requiera que la trama previa completa sea enviada múltiples veces. Dicha información puede ser codificada a la misma velocidad de bits que la trama actual o las previas, o a una velocidad de bits menor. Además, dicha información puede incluir información de tabla de código aleatoria, que aproxima la parte deseada de la señal de excitación, en lugar de una re-codificación completa de la parte deseada de la señal de excitación.

25

Aunque por simplicidad de la representación las operaciones para las diversas técnicas se describen en un orden concreto, secuencial, debe entenderse que este modo de descripción abarca reajustes menores en el orden de las operaciones, salvo que se requiera un orden concreto. Por ejemplo, en algunos casos las operaciones descritas secuencialmente pueden ser reordenadas o llevadas a cabo simultáneamente. Además, por simplicidad, los diagrama de flujo pueden no mostrar las diversas maneras de las cuales pueden utilizarse técnicas concretas junto con otras técnicas.

30

I. Entorno Informático

35 La figura 1 ilustra un ejemplo generalizado de un entorno informático adecuado 100, en el cual pueden implementarse una o más de las realizaciones descritas. El entorno informático 100 no está concebido para sugerir ninguna limitación en cuanto al alcance de la utilización o la funcionalidad de la invención, puesto que la presente invención puede ser implementada en entornos informáticos diversos, de propósito general o especializados.

Haciendo referencia a la figura 1, el entorno informático 100 comprende, por lo menos, una unidad de procesamiento 110 y una memoria 120. En la figura 1, esta configuración básica al máximo 130 está incluida en el interior de una línea de trazos. La unidad de procesamiento 110 ejecuta instrucciones ejecutadas por ordenador, y puede ser un procesador real o un procesador virtual. En un sistema de multi-procesamiento, múltiples unidades de procesamiento ejecutan instrucciones ejecutables por ordenador para incrementar la potencia de procesamiento. La memoria 120 puede ser memoria volátil (por ejemplo registros, caché, RAM) o memoria no volátil (por ejemplo memoria ROM, EEPROM, flash, etc.), o alguna combinación de las dos. La memoria 120 almacena soporte lógico 180 que implementa codificación sub-banda, tablas de código de múltiples etapas y/o técnicas de codificación redundante para un codificador o descodificador de voz.

40

45

Un entorno informático 100 puede tener características adicionales. En la figura 1, el entorno informático 100 incluye almacenamiento 140, uno o más dispositivos de entrada 150, uno o más dispositivos de salida 160, y una o más conexiones de comunicación 170. Un mecanismo de interconexión (no mostrado) tal como un bus, un controlador o una red conecta entre sí los componentes del entorno informático 100. Habitualmente, un soporte lógico (no mostrado) de sistema operativo proporciona un entorno operativo para otro soporte lógico que se ejecuta en el entorno informático 100, y coordina actividades de los componentes del entorno informático 100.

50

El almacenamiento 140 puede ser extraíble o no extraíble, y puede incluir discos magnéticos, cintas magnéticas o casetes, CD-ROMs, CD-RWs, DVDs, o cualquier otro medio que pueda ser utilizado para almacenar información y al que se pueda acceder en el interior del entorno informático 100. El almacenamiento 140 almacena instrucciones para soporte lógico 180.

5 El dispositivo o los dispositivos de entrada 150 pueden ser un dispositivo táctil de entrada tal como un teclado, un ratón, un lápiz, o una bola de seguimiento, un dispositivo de escanear, un adaptador de red, u otro dispositivo que proporcione una entrada al entorno informático 100. Para el audio, el dispositivo o los dispositivos de entrada 150 pueden ser una tarjeta de sonido, un micrófono u otro dispositivo que acepte entradas de audio en forma analógica o digital, o un lector de CD/DVD que proporcione muestras de audio al entorno informático 100. El dispositivo o los
10 dispositivos de salida 160 pueden ser una pantalla, una impresora, un altavoz, un grabador de CD/DVD, un adaptador de red, u otro dispositivo que proporcione salidas desde el entorno informático 100.

La conexión o las 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 legibles por ordenador, información de voz comprimida, u otros datos en una señal de datos modulada. Una señal de datos
15 modulada es una señal que tiene una o más de sus características ajustadas o modificadas de manera que codifica información en la señal. A modo de ejemplo, y no de limitación, los medios de comunicación comprenden técnicas cableadas o inalámbricas implementadas con portadoras eléctricas, ópticas, de RF, infrarrojas, acústicas u otras.

La invención puede describirse en el contexto general de los medios legibles por ordenador. Los medios legibles por ordenador son cualquier medio disponible al que se pueda acceder el interior de un entorno informático. A modo de
20 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 cualquiera de los anteriores.

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

Por simplicidad en la presentación, la descripción detallada utiliza términos como "determinar", "generar", "ajustar" 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 actos realizados por un ser humano. Las operaciones informáticas reales correspondientes a estos términos varían dependiendo de la
30 implementación.

II. Entorno de Red Generalizado y Códec de voz en Tiempo Real

35 La figura 2 es un diagrama de bloques de un entorno de red generalizado 200 junto con el cual pueden ser implementadas una o más de las realizaciones descritas. Una red 250 separa varios componentes del lado del codificador, respecto de varios componentes del lado del descodificador.

Las funciones principales de los componentes del lado del codificador, y del lado del descodificador, son la codificación y la descodificación de voz, respectivamente. En el lado del codificador, una memoria intermedia de
40 entrada 210 acepta y almacena la entrada 202 de voz. El codificador 230 de voz toma la entrada 202 de voz procedente de la memoria intermedia de entrada 210, y la codifica.

En concreto, un divisor de tramas 212 divide las muestras de la entrada de voz 202 en tramas. En una implementación, las tramas son uniformemente de 20 ms de duración (160 muestras para una entrada de 8 kHz y 320 muestras para una entrada de 16 kHz). En otras implementaciones, las tramas tienen duraciones diferentes, son
45 no uniformes o superpuestas, y/o la velocidad de muestreo de la entrada 202 es diferente. Las tramas pueden organizarse en super-trama/trama, trama/sub-trama, u otra configuración para diferentes etapas de la codificación y la descodificación.

Un clasificador de tramas 214 clasifica las tramas de acuerdo con uno o más criterios, tales como la energía de la señal, la frecuencia del paso por cero, la ganancia de predicción a largo plazo, el diferencial de ganancia, y/u otros
50 criterios para sub-tramas o para las tramas completas. Basándose en dichos criterios, el clasificador de tramas 214 clasifica las diferentes tramas en clases tales como silencio, sin voz, con voz, y transición (por ejemplo de sin voz, a con voz). Adicionalmente, las tramas pueden ser clasificadas dependiendo del tipo de codificación redundante, si la hay, que se utiliza para la trama. La clase de trama afecta a los parámetros que se calcularán para codificar la trama. Además, la clase de trama puede afectar a la resolución y a la resistencia a las pérdidas con las cuales se

codifican los parámetros, para proporcionar más resolución y más resistencia a las pérdidas, a parámetros y a clases de trama más importantes. Por ejemplo, las tramas de silencio son codificadas habitualmente a frecuencia muy baja, son muy simples de recuperar por ocultación si se pierden, y pueden no necesitar protección contra pérdida. Las tramas sin voz son codificadas habitualmente a una frecuencia ligeramente superior, son razonablemente sencillas de recuperar por ocultación si se pierden, y no están protegidas significativamente contra pérdidas. Las tramas de voz y de transición son codificadas habitualmente con más bits, dependiendo de la complejidad de la trama así como de la presencia de transiciones. Las tramas con voz y de transición son asimismo difíciles de recuperar si se pierden, y por lo tanto están más significativamente protegidas contra pérdida. Alternativamente, el clasificador de tramas 214 utiliza otras clases de tramas y/o clases de tramas adicionales.

La señal de voz de entrada puede ser dividida en señales sub-banda antes de la aplicación de un modelo de codificación, tal como el modelo de codificación CELP, a la información sub-banda para una trama. Esto puede realizarse utilizando una serie de uno o más bancos de filtros de análisis (tales como los filtros de análisis QMF) 216. Por ejemplo, si va a utilizarse una estructura tribanda, entonces la banda de baja frecuencia puede ser separada haciendo pasar la señal a través de un filtro de paso bajo. Análogamente, la banda alta puede ser separada haciendo pasar la señal a través de un filtro de paso alto. La banda media puede ser separada haciendo pasar la señal a través de un filtro de paso banda, que puede incluir un filtro de paso bajo y un filtro de paso alto en serie. Alternativamente, puede ser utilizados otros tipos de disposiciones de filtro para la descomposición sub-banda y/o la sincronización del filtrado (por ejemplo, antes de la separación de tramas). Si para una parte de la señal va a ser descodificada solamente una banda, esa parte puede puentear los bancos de filtro de análisis 216. La codificación CELP tiene habitualmente una eficiencia de codificación mayor que ADPCM y MLT para señales de voz.

El número de bandas n puede ser determinado por la velocidad de muestreo. Por ejemplo, en una implementación, se utiliza una estructura de una sola banda para una velocidad de muestreo de 8 kHz. Para velocidades de muestreo de 16 kHz y 22,05 kHz, puede utilizarse una estructura de tres bandas tal como se muestra en la figura 3. En la estructura de tres bandas de la figura 3, la banda de baja frecuencia 310 se extiende hasta la mitad del ancho de banda completo F (desde 0 hasta $0,5F$). La otra mitad del ancho de banda está dividida en partes iguales entre la banda media 320 y la banda alta 330. Cerca de las intersecciones de las bandas, la respuesta de frecuencia para una banda puede reducirse gradualmente desde el nivel de paso hasta el nivel de parada, lo que se caracteriza por una atenuación de la señal en ambos lados cuando se aproxima la intersección. Pueden utilizarse asimismo otras divisiones del ancho de banda de frecuencias. Por ejemplo, para la velocidad de muestreo de 32 kHz, puede utilizarse una estructura de cuatro bandas separadas a la misma distancia.

Habitualmente, la banda de baja frecuencia es la banda más importante para señales de voz, debido a que la energía de la señal decae habitualmente hacia los rangos de frecuencias superiores. Por consiguiente, la banda de baja frecuencia se codifica a menudo utilizando más bits que las otras bandas. En comparación con la estructura de codificación de una sola banda, la estructura de sub-bandas es más flexible, y permite un control mejor de la distribución de bits/ruido de cuantificación a través de las bandas de frecuencia. Por consiguiente, se considera que se mejora significativamente la calidad perceptual de la voz utilizando la estructura de sub-bandas.

En la figura 2, cada sub-banda se codifica por separado, tal como se ilustra mediante los componentes de codificación 232, 234. Si bien los componentes 232, 234 de codificación de banda se muestran por separado, la codificación de todas las bandas puede realizarse mediante un solo codificador, o pueden ser codificadas por codificadores diferentes. Dicha codificación de bandas se describe en mayor detalle a continuación, haciendo referencia a la figura 4. Alternativamente, el códec puede funcionar como un códec de una sola banda.

La voz codificada resultante se proporciona al soporte lógico para una o más capas 240 de conexión en red, a través de un multiplexor ("MUX") 236. Las capas de conexión en red 240 procesan la voz codificada para su transmisión sobre la red 250. Por ejemplo, el soporte lógico de las capas de red empaqueta tramas de información de voz codificada, en paquetes que siguen el protocolo RTP, que son transmitidos sobre la red Internet utilizando UDP, IP, y varios protocolos de capa física. Alternativamente, se utilizan capas de soporte lógico diferentes y/o adicionales o protocolos de conexión en red diferentes y/o adicionales. La red 250 es una red de área extendida, por conmutación de paquetes, tal como la red Internet. Alternativamente, la red 250 es una red de área local u otra clase de red.

En el lado del descodificador, el soporte lógico para una o más capas de conexión en red 260 recibe y procesa los datos transmitidos. El soporte lógico y los protocolos de red, de transporte y de las capas superiores, en la capa o las capas de conexión en red 260 del lado del descodificador, se corresponden habitualmente con aquellos de la capa o las capas de conexión en red 240 del lado del codificador. La capa o las capas de conexión en red proporcionan la información de voz codificada al descodificador 270 de voz, a través de un desmultiplexor ("DEMUX") 276. El descodificador 270 descodifica cada una de las sub-bandas por separado, tal como se representa en los módulos de descodificación 272, 274. Todas las sub-bandas pueden ser descodificadas por un solo descodificador, o pueden ser descodificadas por descodificadores de banda diferentes.

A continuación, las sub-bandas descodificadas son sintetizadas en una serie de uno o más bancos de filtros de síntesis (tales como filtros de síntesis QMF) 280, que entregan la voz descodificada 292. Alternativamente, son

utilizados otros tipos de disposiciones de filtro para síntesis sub-banda. Si está presente solamente una banda, entonces la banda descodificada puede puentear los bancos de filtros 280.

La salida de voz descodificada 292 se puede pasar asimismo a través de uno o más filtros posteriores 284 para mejorar la calidad de la salida de voz filtrada resultante 294. Asimismo, cada banda se puede pasar por separado a través de uno o más filtros posteriores antes de entrar en los bancos de filtros 280.

Haciendo referencia a la figura 6, se describe a continuación un descodificador generalizado de banda de voz en tiempo real, pero pueden utilizarse en su lugar otros descodificadores de voz. Adicionalmente, parte o la totalidad de las herramientas y técnicas descritas pueden ser utilizadas con otros tipos de codificadores y descodificadores de audio, tales como codificadores y descodificadores de música, o codificadores y descodificadores de audio de propósito general.

Aparte de estas funciones principales de codificación y descodificación, los componentes pueden asimismo compartir (mostrado en líneas de trazos en la figura 2) información para controlar la velocidad, la calidad y/o la resistencia a pérdidas de voz codificada. El controlador 220 de la velocidad considera una serie de factores tales como la complejidad de la entrada actual en la memoria intermedia de entrada 210, el llenado de memoria intermedia para las memorias intermedias de salida en el codificador 230 o en otro lugar, la velocidad de salida deseada, el ancho de banda actual de la red, las condiciones de congestión/ruido de la red y/o la velocidad de pérdidas del descodificador. El descodificador 270 retroalimenta la información de la velocidad de pérdidas del descodificador, al controlador 220 de la velocidad. La capa o las capas de conexión en red 240, 260 reúnen o estiman información del ancho de banda de la red y de las condiciones de congestión/ruido, la cual es retroalimentada al controlador 220 de la velocidad. Alternativamente, el controlador 220 de la velocidad considera factores diferentes y/o adicionales.

El controlador 220 de la velocidad dirige el codificador de voz 230 para cambiar la frecuencia, la calidad y/o la resistencia a pérdidas, con las cuales es codificada de voz. El codificador 230 puede cambiar la frecuencia y la calidad ajustando factores de cuantificación para parámetros, o cambiando la resolución de los códigos de entropía que representan los parámetros. Adicionalmente, el codificador puede cambiar la resistencia a pérdidas ajustando la frecuencia o el tipo de codificación redundante. Por lo tanto, el codificador 230 puede cambiar la asignación de bits entre las funciones de codificación principales y las funciones de resistencia a las pérdidas, dependiendo de las condiciones de la red.

El controlador 220 de la velocidad puede determinar modos de codificación para cada sub-banda de cada trama basándose en varios factores. Estos factores pueden incluir las características de señal de cada sub-banda, el histórico de la memoria intermedia del flujo de bits, y la velocidad binaria objetivo. Por ejemplo, tal como se ha discutido anteriormente, habitualmente se necesitan menos bits para tramas más simples, tales como tramas sin voz y tramas en silencio, y se necesitan más bits para tramas más complejas, tales como tramas de transición. Adicionalmente, pueden ser necesarios menos bits para algunas bandas, tales como las bandas de alta frecuencia. Además, si la velocidad binaria promedio en la memoria intermedia del histórico del flujo de bits es menor que la velocidad binaria promedio objetivo, puede utilizarse una velocidad binaria superior para la trama actual. Si la velocidad binaria promedio es menor que la velocidad binaria promedio objetivo, entonces puede escogerse una velocidad binaria menor para la trama actual con objeto de reducir la velocidad binaria promedio. Adicionalmente, dichas una o más bandas pueden ser omitidas de una o más tramas. Por ejemplo, las tramas de frecuencia media y alta pueden ser omitidas para tramas sin voz, o pueden ser omitidas para todas las tramas durante un periodo de tiempo con objeto de reducir la velocidad binaria durante ese tiempo.

La figura 4 es un diagrama de bloques de un codificador de banda de voz generalizado 400, con el cual pueden ser implementadas una o más de las realizaciones descritas. El codificador de banda 400 corresponde, en general, a cualquiera de los componentes de codificación de banda 232, 234 de la figura 2.

El codificador de banda 400 acepta la entrada de banda 402 procedente de los bancos de filtros (u otros filtros) si la señal (por ejemplo, la trama actual) está dividida en múltiples bandas. Si la trama actual no está dividida en múltiples bandas, entonces la entrada de banda 400 incluye muestras que representan todo el ancho de banda. El codificador de banda produce la salida de banda codificada 492.

Si una señal está dividida en múltiples bandas, entonces un componente 420 de diezmado puede llevar a cabo el diezmado en cada banda. Como ejemplo, si la velocidad de muestreo se fija a 16 kHz y cada trama dura 20 ms, entonces cada trama incluye 320 muestras. Si no se llevase a cabo diezmado y cada trama fuera dividida en la estructura de tres bandas mostrada en la figura 3, entonces para la trama serían codificadas y descodificadas tres veces el número de muestras (es decir, 320 muestras por banda, o 960 muestras en total). Sin embargo, cada banda puede ser sometida a diezmado. Por ejemplo, la banda de baja frecuencia 310 puede ser diezmada de 320 muestras a 160 muestras, y cada una de la banda media 320 y la banda alta 330 puede ser diezmada de 320 muestras a 80 muestras, donde las bandas 310, 320, 330 se extienden sobre la mitad, un cuarto, y un cuarto del rango de frecuencias, respectivamente (el grado de diezmado 420 en esta implementación varía en relación con el

rango de frecuencia de las bandas 310, 320, 330). No obstante, son posibles otras implementaciones. En las etapas posteriores, habitualmente se utilizan menos bits para las bandas superiores debido a que la energía de la señal habitualmente se reduce hacia los rangos de frecuencias superiores.) Por consiguiente, esto proporciona un total de 320 muestras a codificar y descodificar para la trama.

5 Se considera que incluso con este diezmado de cada banda, el códec de sub-banda puede producir una salida de calidad de la voz superior que un códec de una sola banda, debido a que es más flexible. Por ejemplo, puede ser más flexible controlando el ruido de cuantificación en un esquema por banda, en lugar de utilizar el mismo enfoque para todo el espectro de frecuencias. Cada una de las múltiples bandas pueden ser codificadas con propiedades diferentes (tales como diferentes números y/o diferentes tipos de etapas de tabla de código, tal como se explica a
10 continuación). Dichas propiedades pueden determinarse mediante el control de la velocidad discutido anteriormente, en base a diversos factores, que incluyen la características de señal de cada sub-banda, el histórico de memoria intermedia del flujo de bits y la velocidad binaria objetivo. Tal como se ha discutido anteriormente, habitualmente se requieren menos bits para tramas "simples", tales como tramas sin voz y tramas de silencio, y se requieren más bits para tramas "complejas", tales como tramas de transición. Si la velocidad binaria promedio en la memoria intermedia del histórico del flujo de bits es menor que la velocidad binaria promedio objetivo, puede utilizarse una velocidad binaria superior para la trama actual. De lo contrario, se escoge una velocidad binaria inferior para reducir la velocidad binaria promedio. En un códec sub-banda, cada banda puede caracterizarse de este modo y ser codificada correspondientemente, en lugar de caracterizar todo el espectro de frecuencias del mismo modo. Adicionalmente, el control de la velocidad puede reducir la velocidad binaria mediante omitir una o más de las
20 bandas de frecuencia superior durante una o más tramas.

El componente 430 de análisis LP calcula coeficientes de predicción lineal 432. En una implementación, el filtro LP utiliza diez coeficientes para una entrada de 8 kHz y dieciséis coeficientes para una entrada de 16 kHz, y el componente 430 de análisis LP calcula un conjunto de coeficientes predicción lineal por trama, para cada banda. Alternativamente, el componente 430 de análisis LP calcula dos conjuntos de coeficientes por trama para cada
25 banda, uno para cada una de dos ventanas centradas en posiciones diferentes, o calcula un número diferente de coeficientes por banda y/o por trama.

El componente 435 de procesamiento LPC recibe y procesa los coeficientes de predicción lineal 432. Habitualmente, el componente 435 de procesamiento LPC convierte valores LPC a una representación diferente para una cuantificación y una codificación más eficientes. Por ejemplo, el componente 435 de procesamiento LPC convierte valores LPC a una representación de par espectral lineal ["LSP", line spectral pair], y los valores LSP son cuantificados (tal como por cuantificación vectorial) y codificados. Los valores LSP pueden ser intra-codificados o predichos a partir de otros valores LSP. Son posibles diversas representaciones, técnicas de cuantificación y técnicas de codificación para los valores LPC. Los valores LPC se proporcionan de alguna forma, como parte de la salida 492 de banda codificada para paquetización y transmisión (junto con cualesquiera parámetros de cuantificación y otra información necesaria para la reconstrucción). Para su utilización posterior en el codificador 400, el componente 435 de procesamiento LPC reconstruye los valores LPC. El componente 435 de procesamiento LPC puede llevar a cabo una interpolación para valores LPC (tal como, equivalentemente, en la representación LSP o en otra representación) para suavizar las transiciones entre conjuntos diferentes de coeficientes LPC, o entre los coeficientes LPC utilizados para diferentes sub-tramas o tramas.
30

40 El filtro 440 de síntesis (o "predicción a corto plazo") acepta valores LPC reconstruidos 438 y los incorpora al filtro. El filtro 440 de síntesis recibe una señal de excitación y produce una aproximación de la señal original. Para una trama dada, el filtro 440 de síntesis puede almacenar en memoria intermedia una serie de muestras reconstruidas (por ejemplo, diez para un filtro de diez coeficientes) de la trama anterior, para el comienzo de la predicción.

Los componentes 450, 455 de ponderación perceptual aplican ponderación perceptual a la señal original y a la salida modelada del filtro 440 de síntesis para quitar énfasis selectivamente a la estructura que compone las señales de voz, con objeto de hacer los sistemas auditivos menos sensibles a los errores de cuantificación. Los componentes 450, 455 de ponderación perceptual explotan fenómenos psicoacústicos tales como el enmascaramiento. En una implementación, los componentes 450, 455 de ponderación perceptual aplican ponderaciones en función de los valores LPC originales 432 recibidos desde el componente 430 de análisis LP. Alternativamente, los componentes 450, 455 de ponderación perceptual aplican otras ponderaciones y/o ponderaciones adicionales.
45

A continuación de los componentes 450, 455 de ponderación perceptual, el codificador 400 calcula la diferencia entre la señal original ponderada de manera perceptual y la salida ponderada de manera perceptual del filtro 440 de síntesis, para producir una señal diferencial 434. Alternativamente, el codificador 400 utiliza una técnica diferente para calcular los parámetros de voz.
50

El componente 460 de parametrización de excitación intenta hallar la mejor combinación de índices de tabla de código adaptativo, de índices de tabla de código fija y de índices de tabla de código de ganancia, en términos de minimizar la diferencia entre la señal original ponderada perceptualmente y la señal sintetizada (en términos de error

cuadrático medio ponderado, o de otro criterio). Se calculan muchos parámetros por sub-trama, pero de manera más general los parámetros pueden ser por super-trama, por trama o por sub-trama. Tal como se ha discutido anteriormente, los parámetros para bandas diferentes de una trama o de una sub-trama puede ser diferentes. La tabla 2 muestra los tipos disponibles de parámetros para clases de trama diferentes, en una implementación.

5

Tabla 2: Parámetros para diferentes clases de trama

Clase de trama	Parámetro(s)
Silencio	Información de clase; LSP; ganancia (por trama, para el ruido generado)
Sin voz	Información de clase; LSP; parámetros de tabla de código de impulsos, aleatoria y de ganancia
Con voz	Información de clase; LSP; parámetros de la tabla de código adaptativo, de impulsos, aleatoria y de ganancia (por sub-trama)
Transición	

En la figura 4, el componente 460 de parametrización de excitación divide la trama en sub-tramas y calcula índices de tabla de códigos y ganancias para cada sub-trama, según corresponda. Por ejemplo, el número y el tipo de etapas de tabla de código a utilizar, y las resoluciones de los índices de tabla de código, pueden determinarse inicialmente mediante un modo de codificación, donde el modo puede estar dictado por el componente de control de la velocidad, discutido anteriormente. Un modo concreto puede indicar asimismo parámetros de codificación y descodificación diferentes respecto del número y el tipo de etapas de tabla de código, por ejemplo, la resolución de los índices de la tabla de código. Los parámetros de cada etapa de tabla de código se determinan optimizando los parámetros para minimizar el error entre una señal objetivo y la contribución de dicha etapa de tabla de código a la señal sintetizada. (Tal como se utiliza en el presente documento, el término "optimizar" significa encontrar una solución adecuada bajo limitaciones aplicables tales como la reducción de la distorsión, el tiempo de búsqueda del parámetro, la complejidad de búsqueda del parámetro, la velocidad binaria de los parámetros, etc., frente a llevar a cabo una búsqueda completa en el espacio de parámetros. Análogamente, el término "minimizar" debe entenderse en términos de encontrar una solución adecuada bajo limitaciones aplicables.) Por ejemplo, la optimización puede realizarse utilizando una técnica de error cuadrático medio modificada. La señal objetivo para cada etapa es la diferencia entre la señal residual y la suma de las contribuciones de las etapas de tabla de código anteriores, si las hay, a la señal sintetizada. Alternativamente, pueden utilizarse otras técnicas de optimización.

La figura 5 muestra una técnica para determinar parámetros de tabla de código, de acuerdo con una implementación. El componente 460 de parametrización de la excitación lleva a cabo la técnica, potencialmente junto con otros componentes tales como un controlador de la velocidad. Alternativamente, lleva a cabo la técnica otro componente en un codificador.

Haciendo referencia a la figura 5, para cada sub-trama en una trama de voz o de transición, el componente 460 de parametrización de la excitación determina si puede ser utilizada una tabla de código adaptativa para la sub-trama actual. (Por ejemplo, el control de la velocidad puede imponer que no se va a utilizar ninguna tabla de código adaptativa para una trama concreta.) Si no se va a utilizar la tabla de código adaptativa, entonces un conmutador de tabla de código adaptativa indicará que no van a utilizarse 535 tablas de código adaptativas. Por ejemplo, esto podría realizarse activando un indicador de un bit en el nivel de trama, que indica que no se utilizan tablas de código adaptativas en la trama, mediante especificar un modo de codificación concreto en el nivel de trama, o activando un indicador de un bit para cada sub-trama que indica que no se utiliza ninguna tabla de código adaptativa en la sub-trama.

Por ejemplo, el componente de control de la velocidad puede excluir la tabla de código adaptativa para una trama, eliminando de ese modo la dependencia más significativa con la memoria entre tramas. En concreto para tramas con voz, una señal de excitación típica está caracterizada por un patrón periódico. La tabla de código adaptativa incluye un índice que representa un desfase que indica la posición de un segmento de excitación en la memoria intermedia del histórico. El segmento de excitación anterior se escala, para ser la contribución de la tabla de código adaptativa a la señal de excitación. En el descodificador, la información de la tabla de código adaptativa es, habitualmente, muy significativa en la reconstrucción de la señal de excitación. Si se ha perdido la trama previa y el índice de la tabla de código adaptativa apunta hacia atrás a un segmento de la trama previa, entonces normalmente no es útil el índice de la tabla de código adaptativa, debido a que apunta a una información de histórico inexistente. Incluso si se realizan técnicas de ocultación para recuperar esta información perdida, la reconstrucción futura estará asimismo basada en la señal recuperada de manera imperfecta. Esto provocará que prosiga el error en las tramas siguientes, debido a que la información de desfase es sensible habitualmente.

Por consiguiente, la pérdida de un paquete del que depende una siguiente tabla de código adaptativa, puede conducir a una degradación extendida que se desvanece solamente después de que han sido descodificados muchos paquetes, o cuando se encuentra una trama sin una tabla de código adaptativa. Este problema puede reducirse insertando regularmente las denominadas "intra-tramas" en el flujo de paquetes, que no tienen dependencia de memoria entre tramas. De este modo, los errores se propagarán solamente hasta la siguiente intra-trama. Por consiguiente, existe un compromiso entre una mejor calidad de la voz y una mejor característica de pérdida de paquetes, debido a que la eficiencia de codificación de la tabla de código adaptativa es, habitualmente, superior que la de las tablas de código fijas. El componente de control de la velocidad puede determinar cuándo resulta ventajoso prohibir tablas de código adaptativas para una trama concreta. El conmutador de tabla de código adaptativa puede ser utilizado para impedir la utilización de tablas de código adaptativas para una trama concreta, eliminando de ese modo la que, habitualmente, es la dependencia más significativa con las tramas anteriores (la interpolación LPC y la memoria de filtro de síntesis pueden asimismo depender en alguna medida de las tramas anteriores). Por lo tanto, el conmutador de la tabla de código adaptativa puede ser utilizado por el componente de control de la velocidad para crear una quasi-intra-trama basada dinámicamente en factores tales como la velocidad de pérdidas de paquetes (es decir, cuando la velocidad de pérdidas de paquetes es elevada, pueden ser insertadas más intra-tramas para permitir un reseteo más rápido de la memoria).

También haciendo referencia a la figura 5, si puede utilizarse una tabla de código adaptativa, entonces el componente 460 determina los parámetros de la tabla de código adaptativa. Dichos parámetros incluyen un índice, un valor de paso, que indica un segmento deseado del histórico de la señal de excitación, así como una ganancia para aplicar al segmento deseado. En las figuras 4 y 5, el componente 460 lleva a cabo una búsqueda 520 de paso en bucle cerrado. La búsqueda comienza con el paso determinado por el componente opcional 425 de búsqueda del paso en bucle abierto, de la figura 4. Un componente 425 de búsqueda del paso en bucle abierto analiza la señal ponderada producida por el componente 450 de ponderación, para estimar su paso. Comenzando con este paso estimado, la búsqueda 520 del paso en bucle abierto optimiza el valor de paso para reducir el error entre la señal objetivo y la señal sintetizada ponderada, generada a partir de un segmento indicado del histórico de la señal de excitación. Se optimiza asimismo 525 el valor de ganancia de la tabla de código adaptativa. El valor de ganancia de la tabla de código adaptativa indica un multiplicador a aplicar a los valores de paso predichos (los valores a partir del segmento indicado del histórico de la señal de excitación), para ajustar la escala de los valores. La ganancia multiplicada por los valores de paso predichos es la contribución de la tabla de código adaptativa a la señal de excitación para la trama o sub-trama actual. La optimización 525 de la ganancia produce un valor de ganancia y un valor de índice que minimizan en el error entre la señal objetivo y la señal sintetizada ponderada a partir de la contribución de la tabla de código adaptativa.

Después de que se han determinado los valores de paso y de ganancia, se determina entonces 530 si la contribución de la tabla de código adaptativa es lo suficientemente significativa para que merezca la pena el número de bits utilizados por los parámetros de la tabla de código adaptativa. Si la ganancia de la tabla de código adaptativa es menor que un umbral, la tabla de código adaptativa es desconectada para reservar los bits para las tablas de código fijas discutidas a continuación. En una implementación, se utiliza un valor umbral de 0,3, aunque pueden alternativamente utilizarse como umbral otros valores. A modo de ejemplo, si el modo de codificación actual utiliza la tabla de código adaptativa más una tabla de código de impulsos con cinco impulsos, entonces puede ser utilizada una tabla de código de siete impulsos cuando la tabla de código adaptativa esté desconectada, y el número total de bits seguiría siendo igual o menor. Tal como se ha discutido anteriormente, puede utilizarse un indicador de un bit para cada sub-trama con objeto de indicar el conmutador de tabla de código adaptativa para la sub-trama. Por lo tanto, si no se utiliza la tabla de código adaptativa, se activa el conmutador para indicar que no se utiliza tabla de código adaptativa en la sub-trama 535. Análogamente, si se utiliza la tabla de código adaptativa, se ajusta el conmutador para indicar que se utiliza la tabla de código adaptativa en la sub-trama y se señalizan 540 los parámetros de la tabla de código adaptativa en el flujo de bits. Aunque la figura 5 muestra la señalización después de la determinación, alternativamente, las señales se agrupan en lotes hasta que la técnica finaliza para una trama o super-trama.

El componente 460 de parametrización de la excitación determina asimismo 550 si se utiliza una tabla de código de impulsos. En una implementación, la utilización o no utilización de la tabla de código de impulsos se indica como parte de un modo de codificación global para la trama actual, o puede indicarse o determinarse de otras maneras. Una tabla de código de impulsos es un tipo de tabla de código fija que especifica uno o más impulsos a contribuir a la señal de excitación. Los parámetros de la tabla de código de impulsos comprenden pares de índices y signos (las ganancias pueden ser positivas o negativas). Cada par indica un impulso a incluir en la señal de excitación, el índice indicando la posición del impulso, y el signo indicando la polaridad del impulso. El número de impulsos incluidos en la tabla de código de impulsos y utilizado para contribuir a la señal de excitación, puede variar dependiendo del modo de codificación. Adicionalmente, el número de impulsos puede depender de si está utilizándose o no una tabla de código adaptativa.

Si se utiliza la tabla de código de impulsos, entonces los parámetros de la tabla de código de impulsos son optimizados 555 para minimizar el error entre la contribución de los impulsos indicados y una señal objetivo. Si no se utiliza una tabla de código adaptativa, entonces la señal objetivo es la señal original ponderada. Si se utiliza una

tabla de código adaptativa, entonces la señal objetivo es la diferencia entre la señal original ponderada y la contribución de la tabla de código adaptativa a la señal sintetizada ponderada. En algún punto (no mostrado), los parámetros de la tabla de código de impulsos son entonces señalizados en el flujo de bits.

5 El componente 460 de parametrización de la excitación determina asimismo 565 si van a utilizarse cualesquiera etapas de tabla de código fija aleatoria. El número de etapas (si hay alguna) de tabla de código aleatoria se indica como parte de un modo de codificación global para la trama actual, aunque puede indicarse o determinarse otras maneras. Una tabla de código aleatoria es un tipo de tabla de código fija que utiliza un modelo de señal predefinido para los valores que codifica. Los parámetros de la tabla de código pueden incluir el punto de inicio para un segmento indicado del modelo de señal, y un signo que puede ser positivo o negativo. La longitud o rango del segmento indicado es fijo habitualmente, y por lo tanto no señalizado habitualmente, pero alternativamente se 10 señaliza una longitud o extensión del segmento indicado. Se multiplica una ganancia por los valores en el segmento indicado, para producir la contribución de la tabla de código aleatoria, a la señal de excitación.

15 Si se utiliza por lo menos una etapa de tabla de código aleatoria, entonces son optimizados 570 los parámetros de la etapa de la tabla de código para dicha etapa de la tabla de código, para minimizar el error entre la contribución de la etapa de la tabla de código aleatoria y una señal objetivo. La señal objetivo es la diferencia entre la señal original ponderada y la suma de la contribución, a la señal sintetizada ponderada, de la tabla de código adaptativa (si la hay), de la tabla de código de impulsos (si la hay), y de las etapas (si las hay) de la tabla de código aleatoria determinada previamente. En algún punto (no mostrado), los parámetros de la tabla de código aleatoria son entonces señalizados en el flujo de bits.

20 A continuación, el componente 460 determina 580 si van a utilizarse más etapas de tabla de código aleatoria. En caso afirmativo, entonces los parámetros de la siguiente etapa de tabla de código aleatoria son optimizados 570 y señalizados tal como se ha descrito anteriormente. Esto continúa hasta que han sido determinados todos los parámetros para las etapas de tabla de código aleatoria. Todas las etapas de tabla de código aleatoria pueden utilizar el mismo modelo de señal, aunque probablemente indicarán segmentos diferentes a partir del modelo y 25 tendrán valores de ganancia diferentes. Alternativamente, pueden utilizarse modelos de señal diferentes para diferentes etapas de tabla de código aleatoria.

Cada ganancia de excitación puede ser cuantificada independientemente, o bien pueden ser cuantificadas dos ó mas ganancias conjuntamente, tal como se determine mediante el controlador de la velocidad y/u otros componentes.

30 Si bien en el presente documento ha sido expuesto un orden concreto para optimizar los diversos parámetros de tabla de código, pueden ser utilizados otros órdenes y otras técnicas de optimización. Por lo tanto, aunque la figura 5 muestra el cálculo secuencial de parámetros de tablas de código diferentes, alternativamente se optimizan conjuntamente dos o más parámetros de tablas de código diferentes (por ejemplo, variando conjuntamente los parámetros y evaluando los resultados, de acuerdo con alguna técnica de optimización no lineal). Adicionalmente, 35 podrían utilizarse otras configuraciones de tablas de código u otros parámetros de señal de excitación.

En esta implementación, la señal de excitación es la suma de cualesquiera contribuciones de la tabla de código adaptativa, de la tabla de código de impulsos, y de la etapa o etapas de tabla de código aleatoria. Alternativamente, el componente 460 puede calcular otros parámetros y/o parámetros adicionales para la señal de excitación.

40 Haciendo referencia a la figura 4, los parámetros de tabla de código para la señal de excitación son señalizados o proporcionados de otro modo a un decodificador local 465 (encerrado por líneas de trazos en la figura 4) así como a la salida de banda 492. Por lo tanto, para cada banda, la salida 492 del codificador incluye la salida desde el componente 435 de procesamiento LPC discutido anteriormente, así como la salida desde el componente 460 de parametrización de la excitación.

45 La velocidad binaria de la salida 492 depende, en parte, de los parámetros utilizados por las tablas de código, y el codificador 400 puede controlar la velocidad binaria y/o la calidad conmutando entre conjuntos diferentes de índices de tabla de código, utilizando códigos incrustados, o utilizando otras técnicas. Combinaciones diferentes de etapas y de tipos de tabla de código pueden proporcionar modos de codificación diferentes para diferentes tramas, bandas y/o sub-tramas. Por ejemplo, una trama sin voz puede utilizar solamente una etapa de tabla de código aleatoria. Para una trama de voz de velocidad baja, pueden ser utilizadas una tabla de código adaptativa y una tabla de código de impulsos. Una trama de velocidad alta puede ser codificada utilizando una tabla de código adaptativa, una tabla de código de impulsos, y una o más etapas de tabla de código aleatoria. En una trama, la combinación de todos los modos de codificación para todas las sub-bandas conjuntamente puede denominarse un conjunto de modos. Pueden existir varios conjuntos de modos predefinidos para cada velocidad de muestreo, correspondiendo los diferentes modos a diferentes velocidades binarias de codificación. El módulo de control de la velocidad puede 55 determinar el conjunto de modos para cada trama, o influir sobre el mismo.

El rango de velocidades binarias posibles puede ser muy grande para las implementaciones descritas, y puede producir mejoras significativas en la calidad resultante. En los codificadores estándar, puede asimismo variarse el número de bits que se utiliza para una tabla de código de impulsos, pero demasiados bits pueden simplemente producir impulsos que son demasiado densos. Análogamente, cuando se utiliza solamente una tabla de código, añadir más bits podría permitir la utilización de un modelo de señal mayor. Sin embargo, esto puede incrementar significativamente la complejidad de la búsqueda de segmentos óptimos del modelo. Por contraste, pueden añadirse tipos adicionales de tablas de código y etapas adicionales de tablas de código aleatorias, sin incrementar significativamente la complejidad de las búsquedas de tablas de código individuales (en comparación con buscar una única tabla de código combinada). Además, las etapas múltiples de tablas de código aleatorias y los tipos múltiples de tablas de código fijas permiten factores de ganancia múltiples, lo que proporciona mayor flexibilidad para la adaptación de la forma de onda.

También haciendo referencia a la figura 4, la salida del componente 460 de parametrización de la excitación es recibida por componentes 470, 472, 474, 476 de reconstrucción de tablas de código y por componentes 480, 482, 484, 486 de aplicación de ganancia, en correspondencia con las tablas de código utilizadas por el componente 460 de parametrización. Las etapas 470, 472, 474, 476 de tablas de código y los correspondientes componentes 480, 482, 484, 486 de aplicación de ganancia reconstruyen las contribuciones de las tablas de código. Dichas contribuciones se suman para producir una señal de excitación 490, que es recibida por el filtro de síntesis 440, donde ésta se utiliza junto con las muestras "predichas" a partir de las cuales se produce la predicción lineal subsiguiente. Las partes retardadas de la señal de excitación son utilizadas asimismo como una señal de histórico de excitación por el componente 470 de reconstrucción de tablas de código adaptativas, para reconstruir parámetros de la tabla de código adaptativa subsiguiente (por ejemplo, contribución de paso), y por el componente 460 de parametrización para calcular parámetros de la tabla de código adaptativa subsiguiente (por ejemplo, índice de paso y valores de ganancia de paso).

De nuevo haciendo referencia a la figura 2, la salida de banda para cada banda es aceptada por el MUX 236, junto con otros parámetros. Dichos otros parámetros pueden comprender, entre otra información, información 222 de clase de trama procedente del clasificador 214 de tramas, y modos de codificación de trama. El MUX 236 construye paquetes de la capa de aplicación a pasar a otro soporte lógico, o bien el MUX 236 pone datos en las cargas útiles de los paquetes que siguen un protocolo tal como RTP. El MUX puede almacenar en memoria intermedia parámetros con objeto de permitir la repetición selectiva de los parámetros, para la corrección de errores hacia delante en paquetes posteriores. En una implementación, el MUX 236 empaqueta en un solo paquete la información de voz codificada primaria para una trama, junto con información de la corrección de errores hacia delante para la totalidad o parte de una o más tramas previas.

El MUX 236 proporciona retroalimentación tal como el lleno actual de la memoria intermedia, para propósitos de control de la velocidad. De manera más general, diversos componentes del codificador 230 (que incluyen el clasificador 240 de tramas y el MUX 236) pueden proporcionar información a un controlador 220 de la velocidad, tal como el que se muestra en la figura 2.

El DEMUX 276 del flujo de bits de la figura 2, acepta como una entrada la información de voz codificada, y la analiza sintácticamente para identificar y procesar parámetros. Los parámetros pueden incluir clases de trama, alguna representación de valores LPC, y parámetros de tabla de código. La clase de trama puede indicar qué otros parámetros están presentes para una trama dada. De manera más general, el DEMUX 276 utiliza los protocolos utilizados por el codificador 230 y extrae los parámetros que el codificador 230 empaqueta en paquetes. Para los paquetes recibidos sobre una red dinámica de conmutación por paquetes, el DEMUX 276 incluye una memoria intermedia de inestabilidad para analizar las fluctuaciones de corto plazo en la velocidad de paquetes sobre un periodo de tiempo dado. En algunos casos, el descodificador 270 regula retardos en la memoria intermedia y administra cuándo son leídos los paquetes desde la memoria intermedia con objeto de integrar retardo, control de calidad, ocultación de tramas perdidas, etc., en la descodificación. En otros casos, un componente de la capa de aplicación administra la memoria intermedia de inestabilidad, y la memoria intermedia de inestabilidad se llena a una velocidad variable, y es evacuada por el descodificador 270 a una velocidad constante o relativamente constante.

El DEMUX 276 puede recibir múltiples versiones de parámetros para un segmento dado, que incluyen una versión codificada primaria y una o más versiones secundarias de corrección de error. Cuando la corrección de errores falla, el descodificador 270 utiliza técnicas de ocultación tales como repetición de parámetros o estimación basada en información que se recibió correctamente.

La figura 6 es un diagrama de bloques de un descodificador generalizado 600 de banda de voz en tiempo real, con el cual pueden implementarse una o más realizaciones descritas. El descodificador 600 de banda corresponde, en general, a cualquiera de los componentes 272, 274 de descodificación de banda de la figura 2.

El descodificador 600 de banda acepta, como entrada, información de voz codificada 692 para una banda (que puede ser la banda completa, o una de múltiples sub-bandas), y produce una salida reconstruida 602 después de la descodificación. Los componentes del descodificador 600 tienen componentes correspondientes en el codificador

400, pero globalmente el descodificador 600 es más sencillo puesto que carece de componentes para la ponderación perceptual, el bucle del proceso de excitación y el control de la velocidad.

El componente 635 de procesamiento LPC recibe información que representa valores LPC en la forma prevista por el codificador de banda 400 (así como cualesquiera parámetros de cuantificación y otra información necesaria para la reconstrucción). El componente 635 de procesamiento LPC reconstruye los valores LPC 638 utilizando las inversas de la conversión, la cuantificación, la descodificación, etc., aplicadas previamente a los valores LPC. El componente 635 de procesamiento LPC puede llevar a cabo asimismo interpolación para valores LPC (en representación LPC u otra representación, tal como LSP) para suavizar las transiciones entre conjuntos diferentes de coeficientes LPC.

Las etapas 670, 672, 674, 676 de tabla de código y los componentes 680, 682, 684, 686 de aplicación de ganancia descodifican los parámetros de cualquiera de las etapas correspondientes de tabla de código, utilizados para la señal de excitación, y calculan la contribución de cada etapa de tabla de código que se utiliza. De forma más general, la configuración y las operaciones de las etapas 670, 672, 674, 676 de tabla de código y de los componentes 680, 682, 684, 686 de ganancia, corresponden a la configuración y las operaciones de las etapas 470, 472, 474, 476 de tabla de código y los componentes 480, 482, 484, 486 de ganancia en el codificador 400. Las contribuciones de las etapas de tabla de código utilizadas se suman, y la señal de excitación resultante 690 es alimentada al filtro de síntesis 640. Los valores retardados de la señal de excitación 690 son utilizados asimismo por la tabla de código adaptativa 670 como un histórico de excitación, para calcular la contribución de la tabla de código adaptativa a las partes posteriores de la señal de excitación.

El filtro de síntesis 640 acepta valores LPC reconstruidos 638 y los incorpora al filtro. El filtro de síntesis 640 almacena previamente muestras reconstruidas, para el procesamiento. La señal de excitación 690 se pasa a través del filtro de síntesis para formar una aproximación de la señal de la voz original. De nuevo haciendo referencia a la figura 2, tal como se ha explicado anteriormente, si existen múltiples sub-bandas, la salida de sub-banda para cada sub-banda es sintetizada en los bancos 280 de filtros para formar la salida 292 de voz.

Las relaciones mostradas en las figuras 2 a 6 indican flujos generales de información; no se muestran otras relaciones por simplicidad. Dependiendo de la implementación y del tipo de compresión deseada, pueden añadirse, omitirse, dividirse componentes en múltiples componentes, combinarse con otros componentes y/o sustituirse con componentes similares. Por ejemplo, en el entorno 200 mostrado en la figura 2, el controlador 220 de la velocidad puede combinarse con el codificador 230 de voz. Los componentes añadidos potenciales incluyen una aplicación de codificación (o reproducción) multimedia, que gestiona el codificador (o descodificador) de voz así como otros codificadores (o descodificadores), y reúne información del estado del descodificador y de la red, y que lleva a cabo funciones de corrección de errores adaptativa. En realizaciones alternativas, diferentes combinaciones y configuraciones de componentes, procesan la información de voz utilizando las técnicas descritas en el presente documento.

35 **III. Técnicas de Codificación Redundante**

Una posible utilización de los códecs de voz, es para redes de voz sobre IP u otras redes con conmutación de paquetes. Dichas redes tienen algunas ventajas sobre las infraestructuras existentes de conmutación de circuitos. Sin embargo, en las redes de voz sobre IP, los paquetes a menudo se retardan o se pierden debido a la congestión de la red.

Muchos códecs de voz estándar tienen una dependencia entre tramas elevada. Por lo tanto, para estos códecs, una trama perdida puede provocar una degradación severa de la calidad de la voz a través de muchas tramas posteriores.

En otros códecs, cada trama puede ser descodificada independientemente. Dichos códecs son robustos frente a pérdidas de paquetes. Sin embargo, la eficiencia de la codificación en términos de la calidad y de la velocidad binaria, cae significativamente como resultado de la anulación de la dependencia entre tramas. Por lo tanto, dichos códecs requieren habitualmente velocidades binarias superiores para conseguir una calidad de la voz similar a los códecs CELP tradicionales.

En algunas realizaciones, las técnicas de codificación redundante discutidas a continuación pueden ayudar a conseguir un buen rendimiento de recuperación de la pérdida de paquetes, sin incrementar significativamente la velocidad binaria. Las técnicas pueden ser utilizadas conjuntamente dentro de un solo códec, o pueden ser utilizadas por separado.

En la implementación de codificador descrita anteriormente haciendo referencia a las figuras 2 y 4, la información de la tabla de código adaptativa es habitualmente la fuente principal de dependencia con otras ramas. Tal como se ha discutido anteriormente, el índice de la tabla de código adaptativa indica la posición de un segmento de la señal de

excitación, en la memoria intermedia del histórico. El segmento de la señal de excitación previa se escala (de acuerdo con un valor de ganancia) para ser la contribución de la tabla de código adaptativa, de la señal de excitación de la trama (o sub-trama) actual. Si se pierde un paquete previo que contiene información utilizada para reconstruir la señal de excitación previa codificada, entonces esta información de desfase de la trama (o sub-trama) actual no es útil debido a que apunta a información de histórico inexistente. Puesto que la información de desfase es sensible, esto conduce habitualmente a una degradación extendida de la salida de voz resultante, que se desvanece solamente después de que han sido descodificados muchos paquetes.

Las técnicas siguientes están diseñadas para eliminar, al menos en alguna medida, la dependencia de la señal de excitación actual con información reconstruida a partir de tramas anteriores que no están disponibles debido a que han sido retardadas o perdidas.

Un codificador, tal como el codificador 230 descrito anteriormente haciendo referencia a la figura 2, puede conmutar entre las técnicas de codificación siguientes en una base trama-a-trama o en alguna otra base. Un descodificador correspondiente, tal como el descodificador 270 descrito anteriormente haciendo referencia a la figura 2, conmuta técnicas correspondientes de análisis sintáctico/descodificación en una base trama-a-trama o en alguna otra base. Alternativamente, otro codificador, descodificador, o herramienta de procesamiento de audio lleva a cabo una o más de las técnicas siguientes.

A. Re-Codificación/Descodificación Primaria del Histórico de la Tabla de Código Adaptativa

En la re-codificación/descodificación primaria del histórico de tabla de código adaptativa, la memoria intermedia del histórico de excitación no se utiliza para descodificar la señal de excitación de la trama actual, incluso si está disponible en el descodificador la memoria intermedia del histórico de excitación (paquete de trama anterior recibido, trama anterior descodificada, etc.). En su lugar, en el codificador, se analiza la información de paso para la trama actual, con objeto de determinar qué parte del histórico de excitación se necesita. La parte necesaria del histórico de excitación es re-codificada y enviada junto con la información codificada (por ejemplo, parámetros de filtro, ganancias e índices de tabla de código) para la trama actual. La contribución de la tabla de código adaptativa de la trama actual hace referencia a la señal de excitación re-codificada que es enviada con la trama actual. De este modo, se garantiza que el histórico de excitación relevante está disponible para el descodificador, para cada trama. Esta codificación redundante no es necesaria si la trama actual no utiliza una tabla de código adaptativa, tal como una trama sin voz.

La re-codificación de la parte de referencia del histórico de excitación, puede realizarse junto con la codificación de la trama actual, y puede realizarse del mismo modo que la codificación de la señal de excitación para una trama actual, que se ha descrito anteriormente.

En algunas implementaciones, la codificación de la señal de excitación se realiza en una base de sub-trama, y el segmento de la señal de excitación re-codificada se extiende hacia atrás desde el comienzo de la trama actual que incluye la sub-trama actual, hasta el límite de la sub-trama más allá de la dependencia más lejana de la tabla de código adaptativa, para la trama actual. Por lo tanto, la señal de excitación re-codificada está disponible para referencia, con la información de paso para múltiples sub-tramas en la trama. Alternativamente, la codificación de la señal de excitación se realiza sobre alguna otra base, por ejemplo, trama-a-trama.

En la figura 7 se muestra un ejemplo, que representa un histórico de excitación 710. Los límites de trama 720 y los límites de sub-trama 730 se representan por líneas de trazos más largas y más cortas, respectivamente. Las sub-tramas de una trama actual 740 están codificadas utilizando una tabla de código adaptativa. El punto de dependencia más lejano para cualquier índice de desfase de tabla de código adaptativa, de una sub-trama de la trama actual, está representado por una línea 750. Por consiguiente, el histórico re-codificado 760 se extiende hacia atrás, desde el comienzo de la trama actual hasta el siguiente límite de sub-trama pasado el punto más lejano 750. El punto de dependencia más lejano puede ser estimado utilizando los resultados de la búsqueda 425 de paso en bucle abierto, descrita anteriormente. Sin embargo, puesto que la búsqueda no es precisa, es posible que la tabla de código adaptativa dependa de una parte de la señal de excitación que esté más allá del punto más lejano estimado, salvo que la búsqueda del último paso esté limitada. Por consiguiente, el histórico re-codificado puede incluir muestras adicionales más allá del punto de dependencia más lejano estimado, para proporcionar espacio adicional para encontrar información de paso coincidente. En una implementación, en el histórico re-codificado se incluyen, por lo menos, diez muestras adicionales pasado el punto de dependencia más lejano estimado. Por supuesto, pueden incluirse más de diez muestras, con objeto de incrementar la probabilidad de que el histórico re-codificado se extienda lo suficientemente lejos para incluir ciclos de paso que se correspondan con los de la sub-trama actual.

Alternativamente, son re-codificados solamente el segmento o los segmentos de la señal de excitación anterior, a los que de hecho se hace referencia en la sub-trama o en las sub-tramas de la trama actual. Por ejemplo, un segmento de la señal de excitación anterior con la duración apropiada, es re-codificado para ser utilizado en la descodificación de un solo segmento actual de dicha duración.

5 La re-codificación/descodificación primaria del histórico de la tabla de código adaptativa, elimina la dependencia sobre el histórico de excitación de las tramas anteriores. Al mismo tiempo, permite que sean utilizadas las tablas de código adaptativas y no requiere la re-codificación de toda la trama o las tramas anteriores (o incluso de todo el histórico de excitación de la trama o de las tramas anteriores). Sin embargo, la velocidad binaria necesaria para la re-codificación de la memoria de tabla de código adaptativa es muy elevada en comparación con las técnicas descritas a continuación, especialmente cuando el histórico re-codificado se utiliza para la codificación/descodificación primaria al mismo nivel de calidad que la codificación/descodificación con dependencia entre tramas.

10 Como consecuencia de la re-codificación/descodificación primaria del histórico de la tabla de código adaptativa, la señal de excitación re-codificada puede ser utilizada para recuperar, por lo menos, parte de la señal de excitación para una trama perdida previa. Por ejemplo, la señal de excitación re-codificada es reconstruida durante la descodificación de las sub-tramas de una trama actual, y la señal de excitación re-codificada es introducida en un filtro de síntesis LPC construido utilizando coeficientes de filtro reales o estimados.

15 La señal de salida reconstruida resultante puede ser utilizada como parte de la salida de la trama anterior. Esta técnica puede ayudar asimismo a estimar un estado inicial de la memoria del filtro de síntesis para la trama actual. Utilizando el histórico de excitación re-codificado y la memoria estimada del filtro de síntesis, la salida de la trama actual se genera del mismo modo que en la codificación normal.

B. Re-Codificación/Descodificación Secundaria del Histórico de la Tabla de Código Adaptativa

20 En la re-codificación/descodificación secundaria del histórico de la tabla de código adaptativa, no se modifica la codificación primaria de la tabla de código adaptativa de la trama actual. Análogamente, no se modifica la descodificación primaria de la trama actual; utiliza el histórico de excitación de la trama anterior, si ha sido recibida la trama anterior.

25 Para su utilización, si el histórico de excitación anterior no está reconstruido, la memoria intermedia del histórico de excitación es re-codificada sustancialmente de la misma manera que la técnica de re-codificación/descodificación primaria del histórico de la tabla de código adaptativa, que se ha descrito anteriormente. Sin embargo, en comparación con la re-codificación/descodificación primaria, se utilizan menos bits para la re-codificación debido a que la calidad de la voz no está influida por la señal re-codificada cuando no se han perdido paquetes. El número de bits utilizados para re-codificar el histórico de excitación puede reducirse cambiando varios parámetros, tal como utilizando menos etapas de tabla de código fija, o utilizando menos impulsos en la tabla de código de impulsos.

30 Cuando se ha perdido una trama previa, se utiliza en el descodificador el histórico de excitación re-codificado, para generar la señal de excitación de la tabla de código adaptativa, para la trama actual. El histórico de excitación re-codificado puede ser utilizado asimismo para recuperar, por lo menos, parte de la señal de excitación para una trama previa perdida, tal como en la técnica de re-codificación/descodificación primaria del histórico de la tabla de código adaptativa.

35 Asimismo, la señal de salida reconstruida resultante puede ser utilizada como parte de la salida de la trama previa. Esta técnica puede ayudar asimismo a estimar un estado inicial de la memoria del filtro de síntesis para la trama actual. Utilizando el histórico de excitación re-codificado y la memoria estimada del filtro de síntesis, la salida de la trama actual se genera del mismo modo que en la codificación normal.

C. Etapa de Tabla de Código Extra

40 Tal como en la técnica de re-codificación/descodificación secundaria del histórico de la tabla de código adaptativa, en la técnica de etapa de tabla de código extra, la codificación de la señal de excitación principal es igual que la codificación normal descrita anteriormente haciendo referencia a las figuras 2 a 5. Sin embargo, se determinan asimismo parámetros para una etapa de tabla de código extra.

45 En esta técnica de codificación, que se muestra en la figura 8, se asume 810 que la memoria intermedia del histórico de excitación anterior es todo cero al comienzo de la trama actual, y por lo tanto no existe contribución desde la memoria intermedia del histórico de excitación anterior. Además de la información principal codificada para la trama actual, se utilizan una o más etapas de tabla de código extra para cada sub-trama u otro segmento que utilice una tabla de código adaptativa. Por ejemplo, la etapa de tabla de código extra utiliza una tabla de código fija aleatoria, tal como la que se describe haciendo referencia a la figura 4.

50 En esta técnica, la trama actual está codificada normalmente para producir información principal codificada (la cual puede incluir parámetros de la tabla de código principal para etapas de la tabla de código principal) a utilizar por el descodificador, si la trama previa está disponible. En el lado del codificador, los parámetros redundantes para una o más etapas de tabla de código extra se determinan en el bucle cerrado, asumiendo que no hay información de

excitación procedente de la trama previa. En una primera implementación, la determinación se realiza sin utilizar ninguno de los parámetros de la tabla de código principal. Alternativamente, en una segunda implementación la determinación utiliza, por lo menos, parte de los parámetros de la tabla de código principal para la trama actual. Dichos parámetros de la tabla de código principal pueden ser utilizados junto con el parámetro o los parámetros de la etapa de tabla de código extra, para descodificar la trama actual si la trama previa se ha perdido, tal como se describe a continuación. En general, esta segunda implementación puede conseguir una calidad similar a la primera implementación, utilizándose menos bits para la etapa o las etapas extra de tabla de código.

De acuerdo con la figura 8, la ganancia de la etapa extra de la tabla de código y la ganancia de la última tabla de código de impulso o aleatoria existente, son optimizadas conjuntamente en una búsqueda de bucle cerrado en el codificador, para minimizar el error de codificación. La mayor parte de los parámetros que se generan en codificación normal, son preservados y utilizados en esta optimización. En la utilización, se determina 820 si son utilizadas en codificación normal algunas etapas de tabla de código aleatoria o de impulsos. En caso afirmativo, se optimiza 830 entonces una ganancia revisada de la etapa de la última tabla de código aleatoria o de impulsos existente (tal como la etapa de tabla de código aleatoria n de la figura 4), para minimizar el error entre la contribución de la etapa de tabla de código y una señal objetivo. La señal objetivo para esta optimización es la diferencia entre la señal residual y la suma de las contribuciones de cualesquiera etapas precedentes de tabla de código aleatoria (es decir, todas las etapas precedentes de tabla de código, pero la contribución de tabla de código adaptativa procedente de segmentos de tramas previas se pone a cero).

Los índices y parámetros de ganancia en la etapa de tabla de código aleatoria extra son optimizados 840 de forma similar, para minimizar el error entre la contribución de dicha tabla de código y una señal objetivo. La señal objetivo para la etapa de tabla de código aleatoria extra es la diferencia entre la señal residual y la suma de las contribuciones de la tabla de código adaptativa, de la tabla de código de impulsos (si hay alguna) y de cualesquiera tablas de código aleatorias normales (con la última tabla de código de impulsos o aleatoria normal existente, teniendo la ganancia revisada). La ganancia revisada de la última tabla de código de impulsos o aleatoria normal existente, y la ganancia de la etapa de tabla de código aleatoria extra pueden ser optimizadas por separado, o conjuntamente.

Cuando está en el modo de descodificación normal, el descodificador no utiliza la etapa de tabla de código aleatoria extra, y descodifica una señal de acuerdo con la descripción anterior (por ejemplo, tal como en la figura 6).

La figura 9A ilustra un descodificador sub-banda que puede utilizar una etapa de tabla de código extra cuando un índice de tabla de código adaptativa apunta a un segmento de una trama previa que se ha perdido. En general, el marco es el mismo que el marco de descodificación descrito anteriormente e ilustrado en la figura 6, y las funciones de muchos de los componentes y señales en el descodificador sub-banda 900 de la figura 9, son las mismas que los componentes y señales correspondientes de la figura 6. Por ejemplo, se recibe la información 992 sub-banda codificada, y el componente 935 de procesamiento LPC reconstruye los coeficientes de predicción lineal 938 utilizando esa información, y alimenta los coeficientes al filtro de síntesis 940. Sin embargo, cuando la trama previa se pierde, un componente 996 de reseteo señala un componente 994 de histórico cero, para poner el histórico de excitación a cero para la trama perdida, y alimenta dicho histórico a la tabla de código adaptativa 970. La ganancia 980 es aplicada a la contribución de la tabla de código adaptativa. Por lo tanto, la tabla de código adaptativa 970 tiene una contribución cero cuando su índice apunta a la memoria intermedia del histórico, para la trama perdida, pero puede tener alguna contribución distinta de cero cuando su índice apunta a un segmento en el interior de la trama actual. Las etapas 972, 974, 976 de tabla de código fija aplican sus índices normales recibidos con la información de sub-banda 992. De forma similar, los componentes 982, 964 de ganancia de la tabla de código fija, excepto el componente 986 de ganancia de la última tabla de código normal, aplican sus ganancias normales para producir sus contribuciones respectivas a la señal de excitación 990.

Si está disponible una etapa 988 de tabla de código aleatoria extra, y se ha perdido la trama previa, entonces un componente 996 de reseteo señala un conmutador 998 para que pase la contribución de la etapa 976 de la última tabla de código normal con una ganancia revisada 987, a sumar con otras contribuciones de tabla de código, en lugar de pasar para suma la contribución de la etapa 976 de la última tabla de código normal con la ganancia normal 986. La ganancia revisada está optimizada para la situación en la que el histórico de excitación está a cero para la trama previa. Adicionalmente, la etapa 978 de tabla de código extra aplica su índice para indicar en la tabla de código correspondiente un segmento de la señal del modelo de tabla de código aleatoria, y el componente 988 de ganancia de la tabla de código aleatoria aplica a dicho segmento la ganancia para la etapa de tabla de código aleatoria extra. El conmutador 998 pasa la contribución resultante de la etapa de tabla de código extra, para ser sumada con las contribuciones de las etapas previas 970, 972, 974, 976 de tabla de código, para producir la señal de excitación 990. Por consiguiente, la información redundante para la etapa de tabla de código aleatoria extra (tal como la ganancia y el índice de etapa extra) y la ganancia revisada de la etapa de la última tabla de código aleatoria principal (utilizada en lugar de la ganancia normal para la etapa de la última tabla de código aleatoria principal) son utilizadas para un reajuste rápido de la trama actual a un estado conocido. Alternativamente, se utiliza la ganancia normal para la etapa de la última tabla de código aleatoria principal y/o son utilizados algunos otros parámetros para señalar una tabla de código aleatoria de una etapa extra.

La técnica de etapa de tabla de código extra requiere tan pocos bits que el coste por su utilización en la velocidad binaria, es prácticamente insignificante. Por otra parte, puede reducir significativamente la degradación de la calidad debida a pérdidas de trama, cuando se presentan dependencias entre tramas.

5 La figura 9B ilustra un descodificador sub-banda similar al ilustrado en la figura 9A, pero sin etapas de tabla de código aleatoria normal. Por lo tanto, en esta implementación, la ganancia revisada 987 está optimizada para la tabla de código de impulsos 972 cuando el histórico residual para una trama previa perdida está a cero. Por consiguiente, cuando se ha perdido una trama, se suman las contribuciones de la tabla de código adaptativa 970 (con el histórico residual para la trama perdida previa puesto a cero), de la tabla de código de impulsos 972 (con la ganancia revisada) y de la etapa 978 de tabla de código aleatoria extra, para producir una señal de excitación 990.

10 Una tabla de código de una etapa extra, que está optimizada para la situación en la que se ha puesto a cero el histórico residual para una trama perdida, puede ser utilizada con muchas implementaciones y combinaciones diferentes de tabla de código y/u otras representaciones de señales residuales.

D. Compromisos entre Técnicas de Codificación Redundante

15 Cada una de las tres técnicas de codificación redundante discutidas anteriormente pueden tener ventajas y desventajas, comparadas con las otras. La tabla 3 muestra algunas conclusiones generalizadas sobre lo que se cree son algunos de los compromisos entre estas tres técnicas de codificación redundante. La penalización sobre la velocidad binaria se refiere a la cantidad de bits que son necesarios para utilizar la técnica. Por ejemplo, asumiendo que se utiliza la misma velocidad binaria que en la codificación/descodificación normal, la penalización superior de la velocidad binaria se corresponde, en general, con una calidad menor durante la descodificación normal, debido a
 20 que se utilizan más bits para la codificación redundante y, por lo tanto, pueden ser utilizados menos bits para la información codificada normal. La eficiencia de la reducción de la dependencia con la memoria se refiere a la eficiencia de la técnica para mejorar la calidad de la salida de voz resultante, cuando se han perdido una o más tramas previas. La utilidad para la recuperación de la trama o las tramas previas, se refiere a la capacidad de utilizar la información codificada de forma redundante, para recuperar dichas una o más tramas previas, cuando la trama o
 25 las tramas previas se han perdido. Las conclusiones de la tabla son generalizadas, y pueden no aplicar en implementaciones concretas.

Tabla 3: Compromisos entre Técnicas de Codificación Redundante

	Codificación Primaria del Histórico de ACB	Codificación Secundaria del Histórico de ACB	Etapas de Tabla de Código Extra
Penalización sobre la velocidad binaria	Alta	Media	Baja
Eficiencia de la reducción de la dependencia con la memoria	Máxima	Buena	Muy buena
Utilidad para la recuperación de la trama o las tramas previas pérdidas	Buena	Buena	Ninguna

30 El codificador puede elegir cualquiera de los esquemas de codificación redundante para cualquier trama, sobre la marcha, durante la codificación. La codificación redundante puede no ser utilizada en absoluto para algunas clases de tramas (por ejemplo, utilizada para tramas de voz, no utilizada para tramas de silencio o sin voz), y si se utiliza puede serlo sobre cada trama, sobre un esquema periódico tal como cada diez tramas, o sobre algún otro esquema. Esto puede estar controlado por un componente tal como el componente de control de la velocidad, considerando factores tales como los compromisos anteriores, el ancho de banda disponible del canal, y la retroalimentación del descodificador sobre el estado de pérdida de paquetes.

35 **E. Formato del Flujo de Bits en la Codificación Redundante**

La información de codificación redundante puede ser enviada en diversos formatos diferentes, en un flujo de bits. Lo que sigue es una implementación de un formato para enviar la información codificada redundante descrita anteriormente, y señalar su presencia a un descodificador. En esta implementación, cada trama de flujo de bits es iniciada con un campo de dos bits denominado tipo de trama. El tipo de trama se utiliza para identificar el modo de

codificación redundante para los bits siguientes, y puede ser utilizado asimismo para otros propósitos en la codificación y la decodificación. La tabla 4 proporciona el significado del modo de codificación redundante, para el campo tipo de trama.

Tabla 4: Descripción de los Bits de Tipo de Trama

Bits de Tipo de Trama	Modo de Codificación Redundante
00	Ninguno (Trama Normal)
01	Etapas de Tabla de Código Extra
10	Codificación Primaria del Histórico de ACB
11	Codificación Secundaria del Histórico de ACB

5

La figura 10 muestra cuatro combinaciones diferentes de estos códigos en el formato de trama del flujo de bits, señalizando la presencia de una trama normal y/o de los tipos de codificación redundante respectivos. Para una trama normal 1010 que incluye información principal codificada para la trama, sin ningún bit de codificación redundante, un límite de octeto 1015 al comienzo de la trama está seguido por un código de tipo de trama 00. Para una trama normal, el código del tipo de trama está seguido por la información principal codificada.

10

Para una trama 1020 con información codificada redundante del histórico de la tabla de código adaptativa primaria, un límite de octeto 1025 al comienzo de la trama está seguido por el código de tipo de trama 10, que señala la presencia de información del histórico de la tabla de código adaptativa primaria para la trama. El código de tipo de trama está seguido por una unidad codificada para una trama con información principal codificada e información del histórico de la tabla de código adaptativa.

15

Cuando para una trama 1030 está incluida la información codificada redundante del histórico secundario, un límite de octeto 1035 al comienzo de la trama está seguido por una unidad codificada que incluye un código de tipo de trama 00 (el código para una trama normal) seguido por información principal codificada para una trama normal. Sin embargo, después del límite de octeto 1045 al término de la información principal codificada, otra unidad codificada incluye un código de tipo de trama 11 que indica que seguirá información del histórico secundario opcional 1040 (en lugar de la información principal codificada para una trama). Debido a que la información de histórico secundario 1040 se utiliza solamente si se ha perdido la trama previa, un empaquetador u otro componente puede estar dotado de la opción de omitir la información. Esto puede realizarse por varias razones, tal como cuando es necesario reducir la velocidad binaria global, la velocidad de pérdidas de paquetes es baja, o la trama previa está incluida en un paquete con la trama actual. O bien, un demultiplexor u otro componente puede ser dotado de la opción de saltarse la información del histórico secundario cuando se recibe satisfactoriamente la trama normal 1030.

20

25

Análogamente, cuando para una trama 1050 se incluye información codificada redundante de una etapa de tabla de código extra, un límite de octeto 1055 al comienzo de una unidad codificada está seguido por un código de tipo de trama 00 (el código para una trama normal) seguido por información principal codificada para una trama normal. Sin embargo, después del límite de octeto 1065 al término de la información principal codificada, otra unidad codificada incluye un código de tipo de trama 01 que indica que seguirá información 1060 de etapa de tabla de código extra opcional. Tal como con la información del histórico secundario, la información 1060 de etapa de tabla de código extra se utiliza solamente si se ha perdido la trama previa. Por consiguiente, tal como con la información de histórico secundario, un empaquetador u otro componente puede estar dotado de la opción de omitir la información de la etapa de tabla de código extra, o bien un demultiplexor u otro componente puede estar dotado de la opción de saltarse la información de la etapa de tabla de código extra.

30

35

Una aplicación (por ejemplo, una aplicación que maneje la paquetización en la capa de transporte) puede decidir combinar conjuntamente múltiples tramas para formar un paquete mayor con objeto de reducir los bits extra requeridos para las cabeceras de paquete. Dentro del paquete, la aplicación puede determinar los límites de trama mediante explorar el flujo de bits.

40

La figura 11 muestra un posible flujo de bits de un solo paquete 1100 con cuatro tramas 1110, 1120, 1130, 1140. Puede asumirse que serán recibidas todas las tramas en el único paquete si lo son cualesquiera de ellas (es decir, no hay corrupción parcial de datos), y que el desfase de la tabla de código adaptativa, o paso, es habitualmente menor que la longitud de la trama. En este ejemplo, cualquier información de codificación redundante opcional para la trama 2, 1120, para la trama 3, 1130, y para la trama 4, 1140, típicamente no se utilizaría debido a que la trama

45

5 previa estaría siempre presente si lo está la trama actual. Por consiguiente, puede eliminarse la información de codificación redundante opcional para todas las tramas excepto para la primera, en el paquete 1100. Esto tiene como resultado el paquete condensado 1150, en el que la trama 1, 1160, incluye información de la etapa de tabla de código extra opcional, pero toda la información de codificación redundante opcional ha sido eliminada de las tramas restantes 1170, 1180, 1190.

10 Si el descodificador está utilizando la técnica de codificación redundante del histórico primario, una aplicación no perderá ninguno de dichos bits cuando empaqueta tramas conjuntamente en un solo paquete, debido a que se utiliza la información de codificación redundante del histórico primario, se haya perdido o no la trama previa. Sin embargo, la aplicación podría forzar al codificador a codificar dicha trama como una trama normal, si sabe que la trama estará en un paquete de múltiples tramas, y que no será la primera trama en dicho paquete.

15 Aunque las figuras 10 y 11 y la descripción anexa muestran límites alineados con octetos entre tramas y tipos de información, alternativamente los límites pueden no estar alineados con octetos. Además, las figuras 10 y 11 y la descripción anexa muestran códigos de tipo de trama ejemplares, y combinaciones de tipos de trama. Alternativamente, un codificador y un descodificador utilizan otros tipos de trama y/o tipos de trama adicionales, o combinaciones de tipos de trama.

20 Habiéndose descrito e ilustrado los principios de nuestra invención haciendo referencia a las realizaciones descritas, se reconocerá que las realizaciones descritas pueden ser modificadas en su disposición y en sus detalles, sin apartarse de dichos principios. Se comprenderá que los programas, procesos o métodos descritos en el presente documento no están relacionados con, o limitados por ningún tipo concreto de entorno informático, salvo que se indique lo contrario. Diversos tipos de entornos informáticos de propósito general o especializados, pueden ser utilizados con las enseñanzas descritas en el presente documento, o llevar a cabo operaciones acordes con éstas. Los elementos de las realizaciones descritas mostrados en soporte lógico pueden ser implementados en equipamiento físico, y viceversa.

REIVINDICACIONES

1. Un flujo de bits que representa una señal de audio, que comprende información principal codificada para una trama actual (740) que hace referencia a un segmento de una trama previa; e
- 5 información codificada redundante (760),
- caracterizado porque**
- se selecciona el segmento de una trama previa para ser utilizado en la decodificación de la trama actual; y
- la información codificada redundante es para decodificar la trama actual, la información codificada redundante comprendiendo información del histórico de señal asociada con el segmento de referencia de la trama previa.
- 10 2. El flujo de bits de la reivindicación 1, en el que la información principal codificada y la información redundante son codificadas de acuerdo con una técnica de codificación.
3. El flujo de bits de la reivindicación 1, en el que la información del histórico de señal comprende un histórico de excitación para el segmento de referencia, pero no un histórico de excitación para uno o más segmentos a los que no se hace referencia, de la trama previa.
- 15 4. El flujo de bits de la reivindicación 1, en el que la información del histórico de señal se codifica a un nivel de calidad ajustado dependiendo, por lo menos en parte, de la probabilidad de utilización de la información codificada redundante, en la decodificación de la trama actual.
5. Un método para el procesamiento de un flujo de bits que representa una señal de audio, en una herramienta de procesamiento de audio, que comprende la etapa de:
- 20 - codificar información principal para una trama actual que hace referencia a un segmento de una trama previa a utilizar en la decodificación de la trama actual; y además **caracterizado por** las etapas de:
- codificar información redundante para decodificar la trama actual, la información codificada redundante comprendiendo información del histórico de señal asociada con el segmento de referencia de la trama previa; y
- 25 - entregar un resultado.
6. El método de la reivindicación 5, en el que la información principal y la información redundante son codificadas de acuerdo con una técnica de codificación.
7. El método de la reivindicación 5, en el que la herramienta de procesamiento de audio es un codificador de voz en tiempo real, y el resultado es la voz codificada.
- 30 8. El método de la reivindicación 5, en el que la información del histórico de señal comprende un histórico de excitación para el segmento de referencia, pero no un histórico de excitación para uno o más segmentos a los que no se hace referencia, de la trama previa.
9. El método de cualquiera de las reivindicaciones 5 a 8, en el que la información del histórico de señal se codifica a un nivel de calidad ajustado dependiendo, por lo menos en parte, de la probabilidad de utilización de la información codificada redundante, en la decodificación de la trama actual.
- 35 10. Un método para el procesamiento de un flujo de bits que representa una señal de audio, en una herramienta de procesamiento de audio, que comprende las etapas de:
- 40 - decodificar información principal codificada, para una trama actual que hace referencia a un segmento de una trama previa a utilizar en la decodificación de la trama actual; y además **caracterizado por** la etapa de:
- decodificar información codificada redundante, para decodificar la trama actual, la información codificada redundante comprendiendo información del histórico de señal asociada con el segmento de referencia de la trama previa; y

- entregar un resultado.

11. El método de la reivindicación 10, en el que la información principal codificada y la información codificada redundante, son codificadas de acuerdo con una técnica de codificación.
- 5 12. El método de la reivindicación 10, en el que la herramienta de procesamiento de audio es un descodificador de voz, y en el que el procesamiento comprende utilizar la información codificada redundante en la descodificación de la trama actual esté, o no, disponible para el descodificador la trama previa.
13. El método de la reivindicación 10, en el que la herramienta de procesamiento de audio es un descodificador de voz, y en el que el procesamiento comprende utilizar la información codificada redundante, en la descodificación de la trama actual, solamente si la trama previa no está disponible para el descodificador.
- 10 14. El flujo de bits de la reivindicación 10, en el que la información del histórico de señal se codifica a un nivel de calidad ajustado dependiendo, por lo menos en parte, de la probabilidad de utilización de la información codificada redundante, en la descodificación de la trama actual.
- 15 15. El método de la reivindicación 10, en el que la herramienta de procesamiento de audio es un descodificador de voz, y en el que el procesamiento comprende utilizar la información codificada redundante en la descodificación de la trama previa, cuando la trama previa no está disponible para el descodificador.

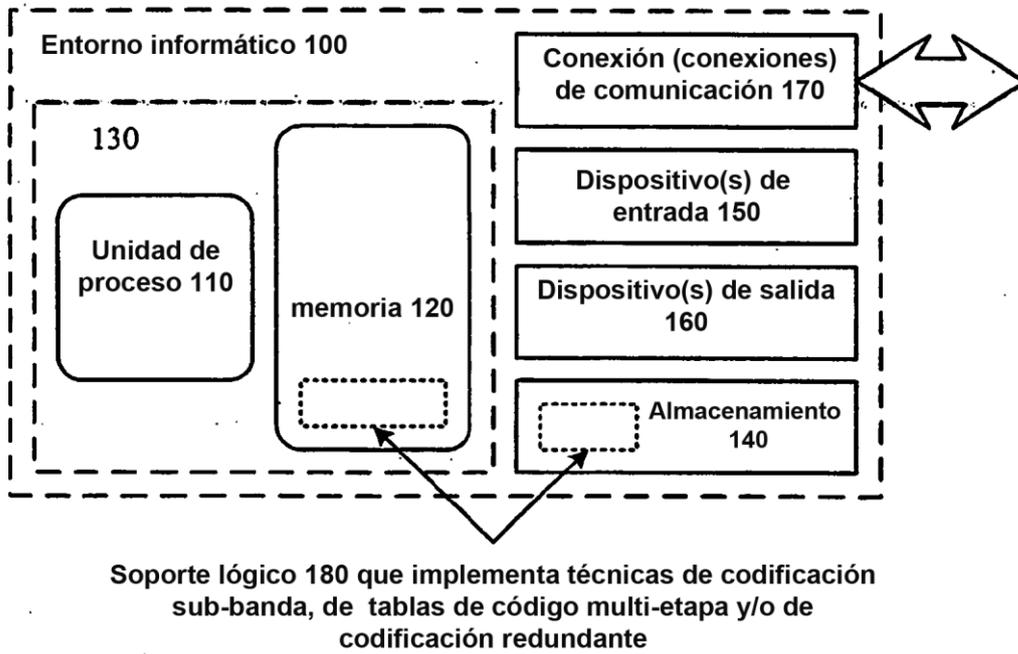


Figura 1

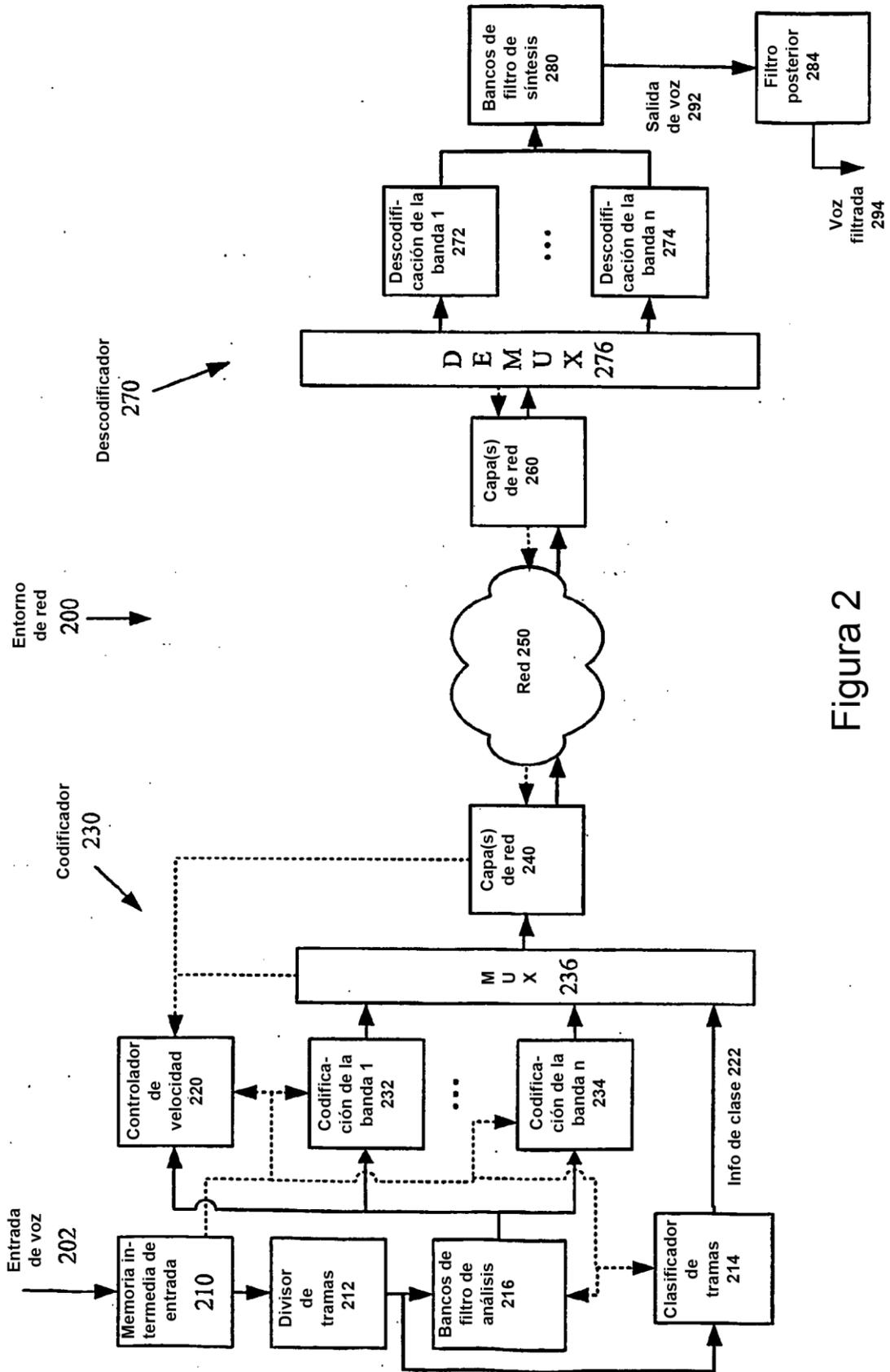


Figura 2

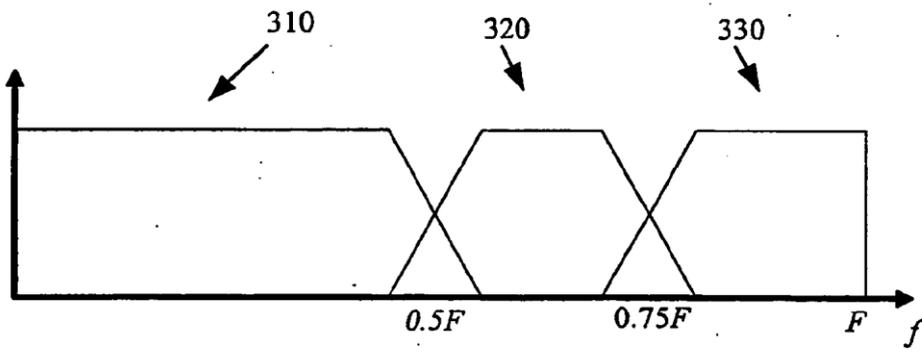
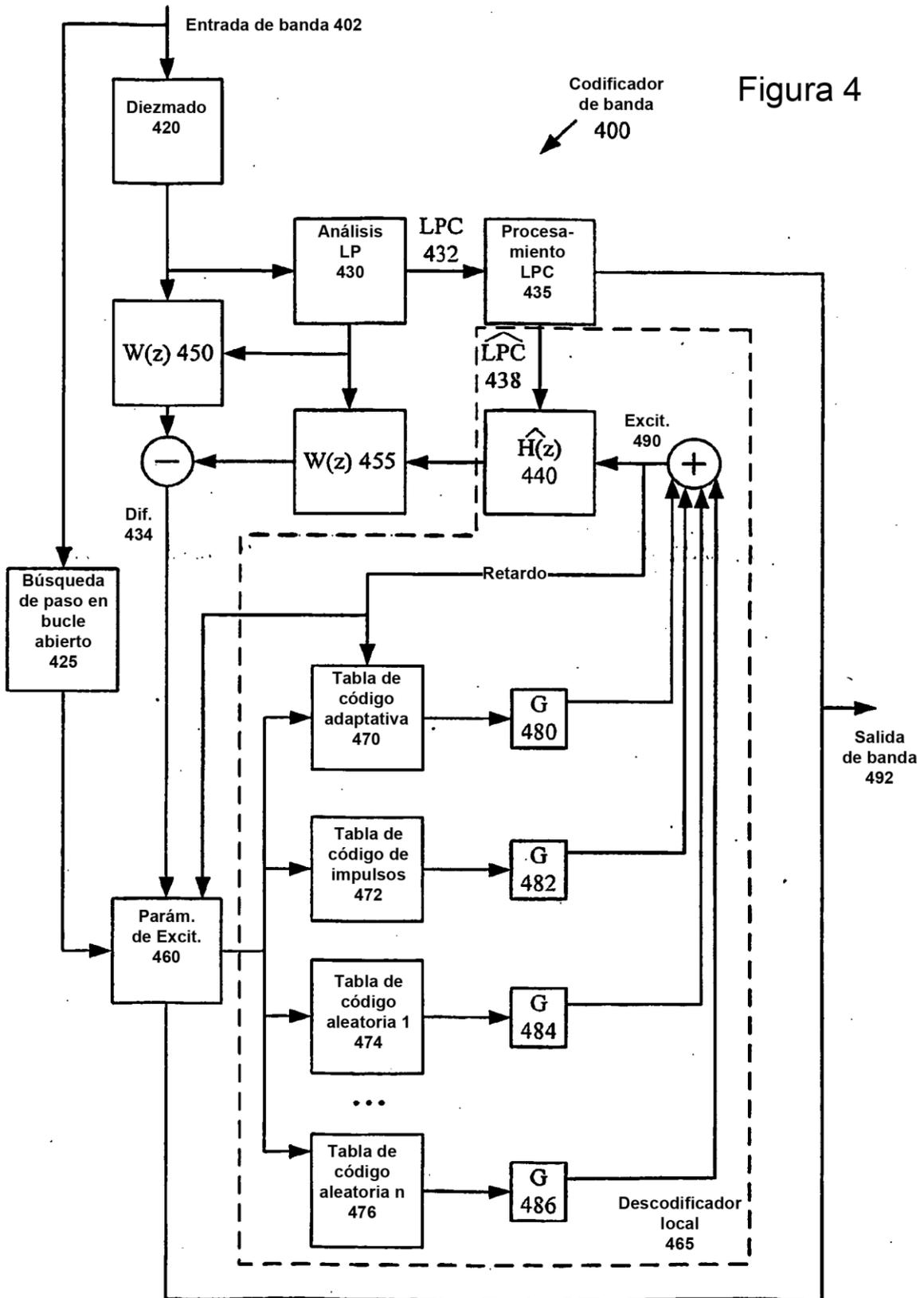


Figura 3

Figura 4



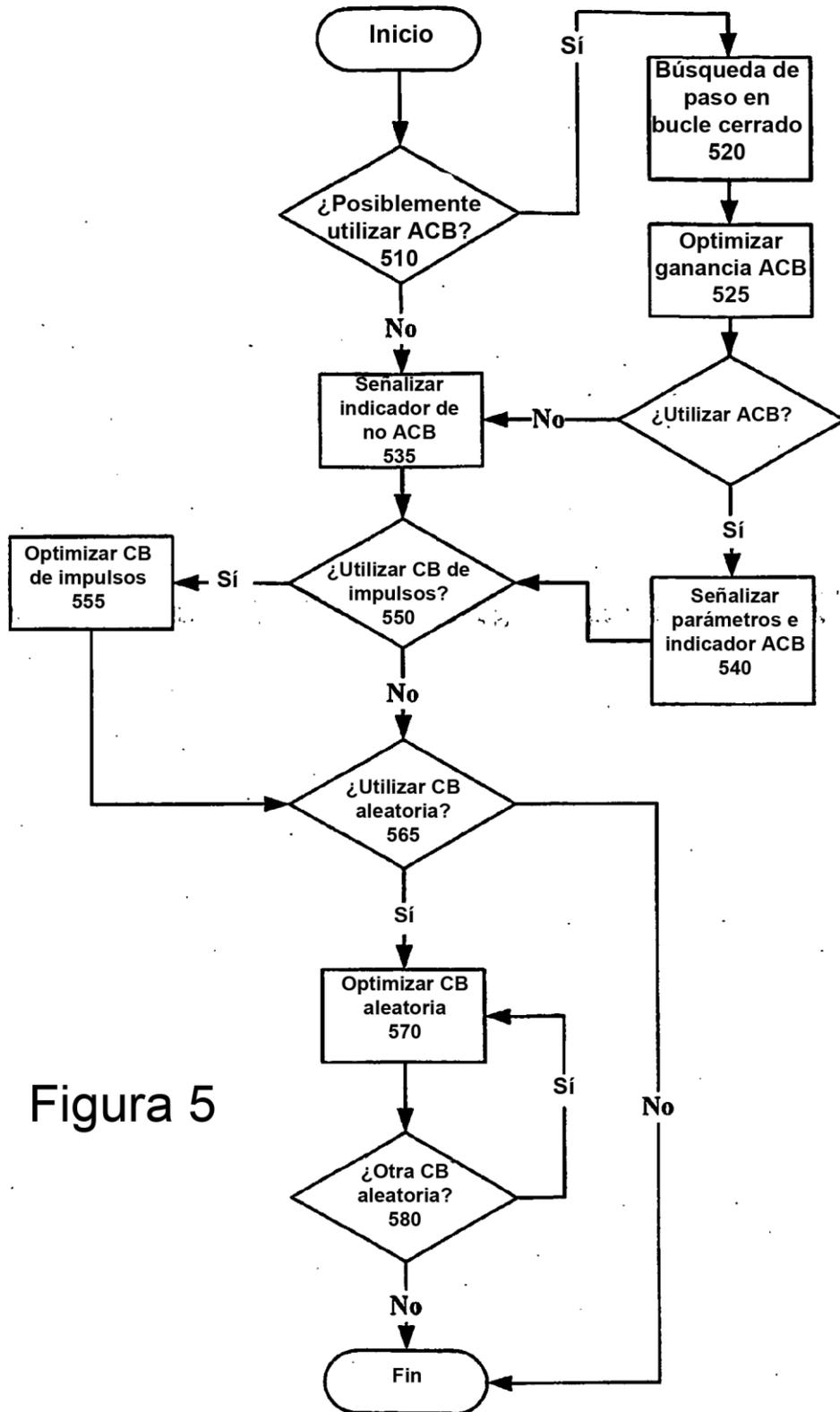
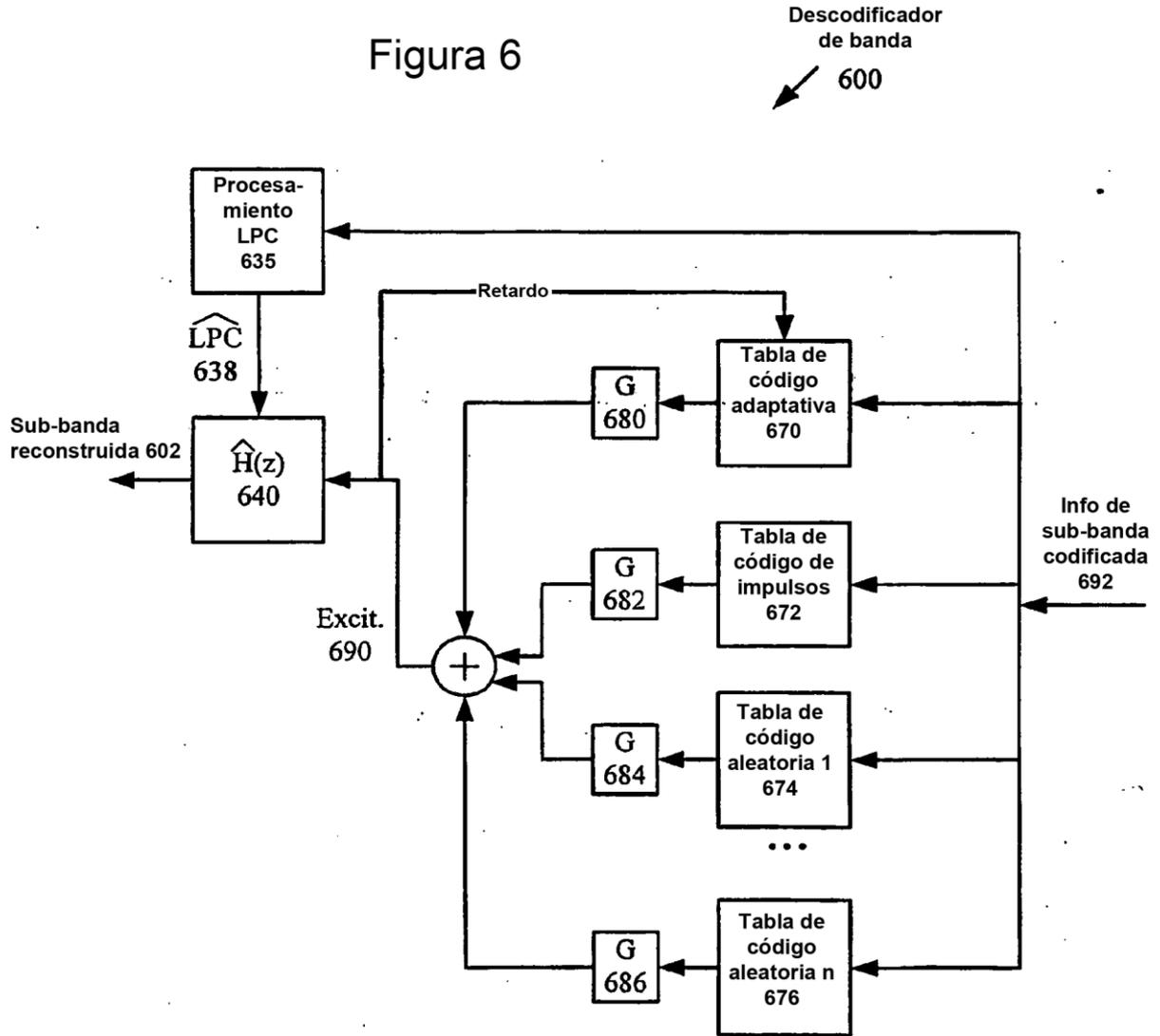


Figura 5

Figura 6



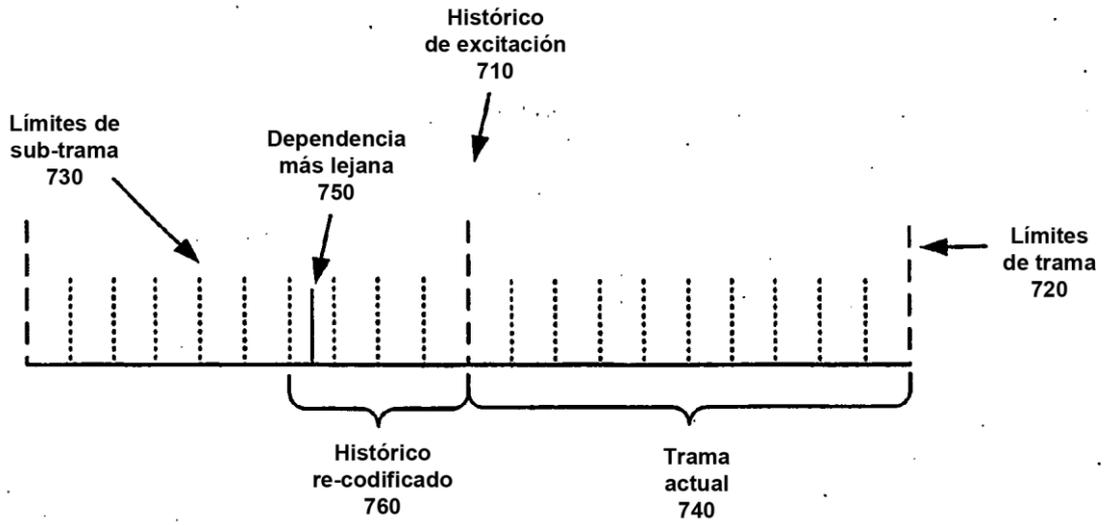


Figura 7

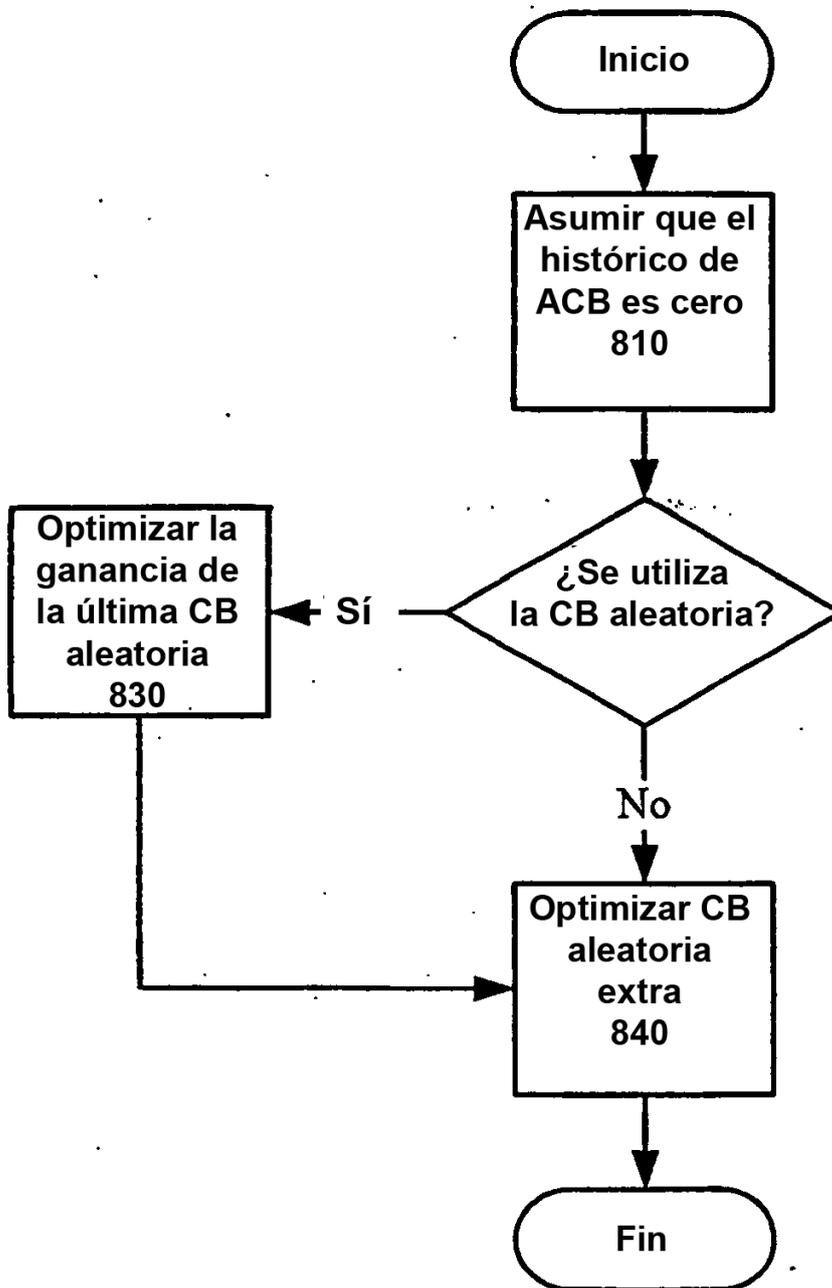
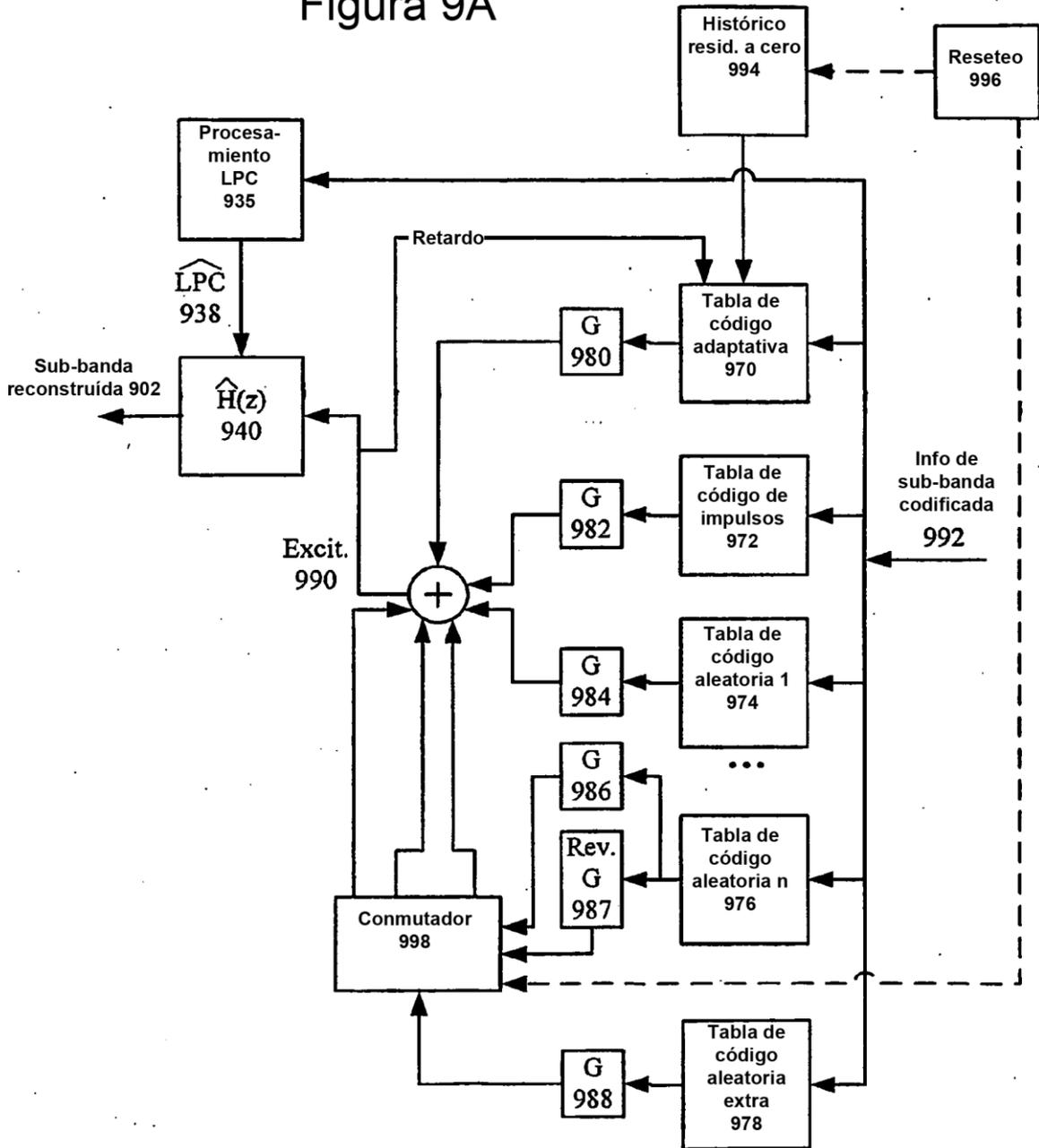


Figura 8

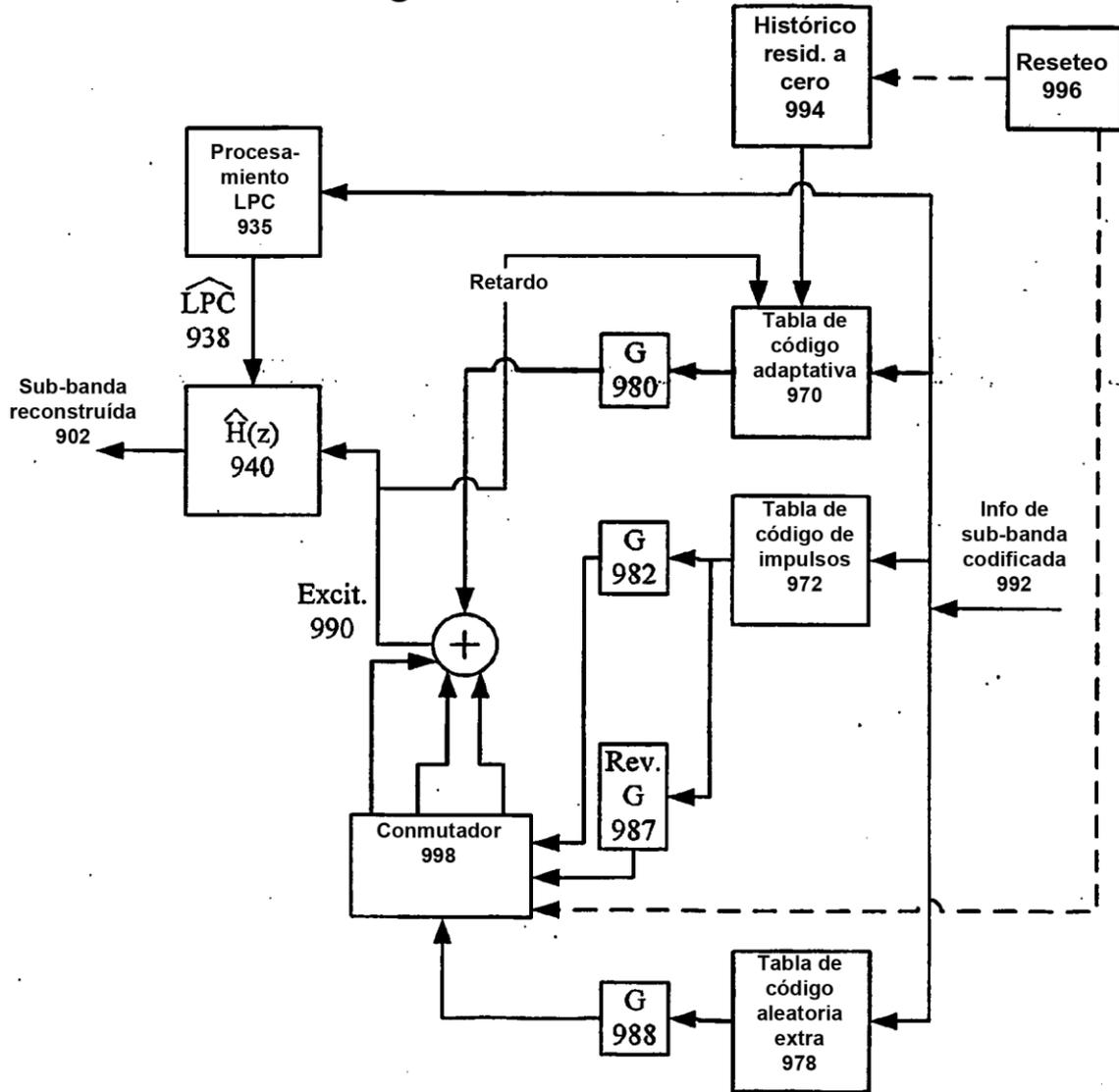
Descodificador de banda
900

Figura 9A



Descodificador de banda
900

Figura 9B



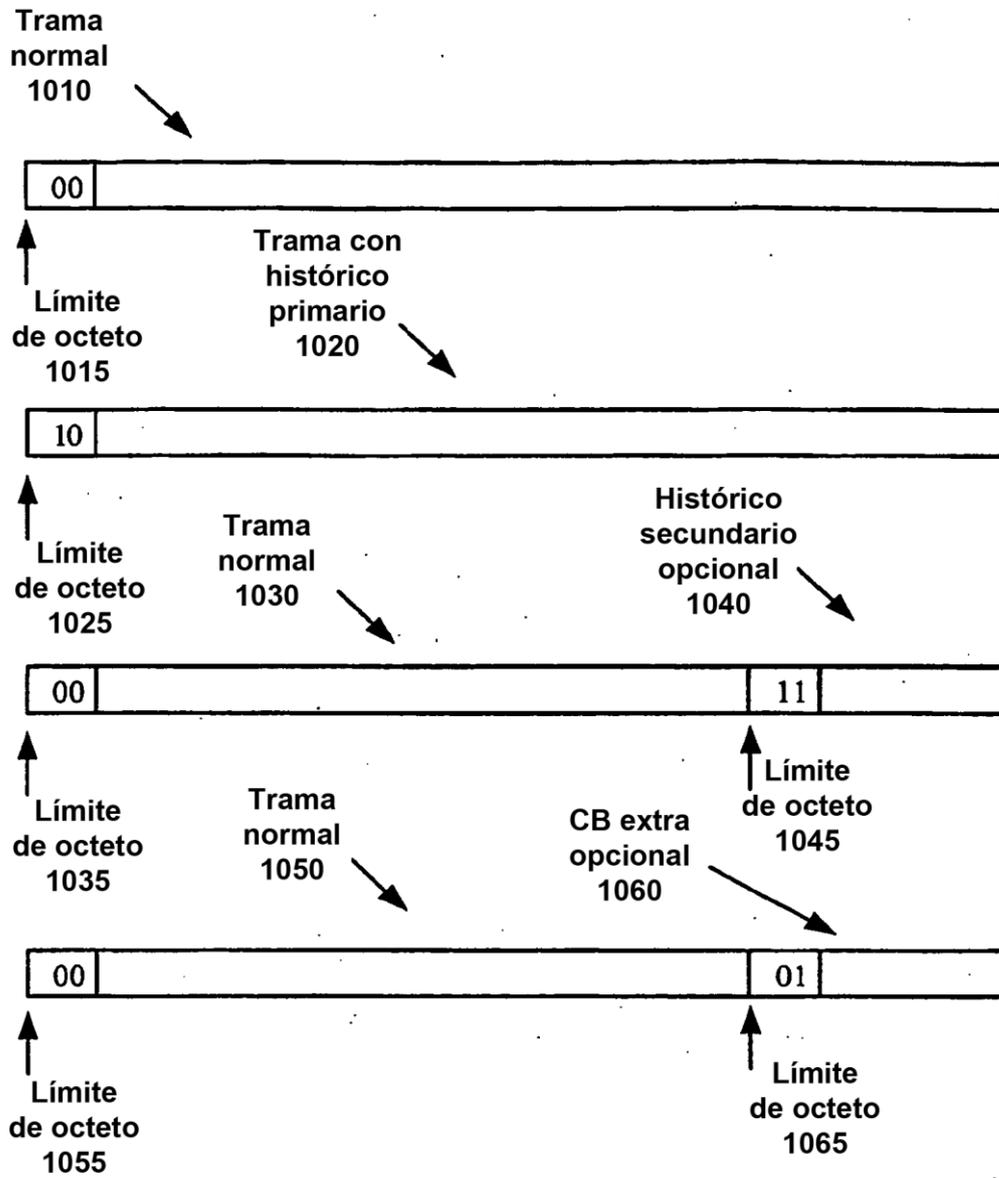


Figura 10

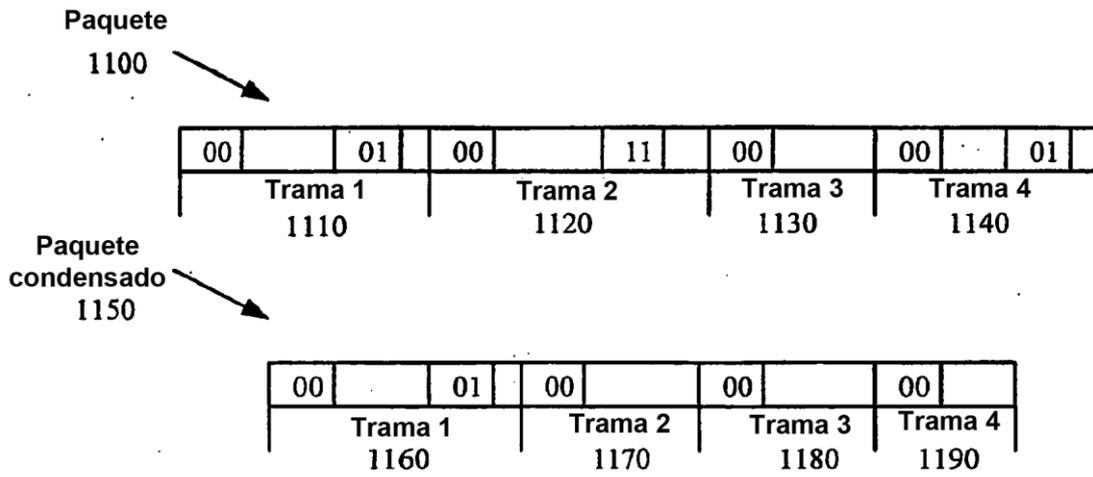


Figura 11