



UNIÓN INTERNACIONAL DE TELECOMUNICACIONES

UIT-T

SECTOR DE NORMALIZACIÓN
DE LAS TELECOMUNICACIONES
DE LA UIT

G.729

(03/96)

**ASPECTOS GENERALES DE LOS SISTEMAS
DE TRANSMISIÓN DIGITAL**

**CODIFICACIÓN DE LA VOZ A 8 kbit/s
MEDIANTE PREDICCIÓN LINEAL CON
EXCITACIÓN POR CÓDIGO ALGEBRAICO
DE ESTRUCTURA CONJUGADA**

Recomendación UIT-T G.729

(Anteriormente «Recomendación del CCITT»)

PREFACIO

El UIT-T (Sector de Normalización de las Telecomunicaciones) es un órgano permanente de la Unión Internacional de Telecomunicaciones (UIT). Este órgano estudia los aspectos técnicos, de explotación y tarifarios y publica Recomendaciones sobre los mismos, con miras a la normalización de las telecomunicaciones en el plano mundial.

La Conferencia Mundial de Normalización de las Telecomunicaciones (CMNT), que se celebra cada cuatro años, establece los temas que han de estudiar las Comisiones de Estudio del UIT-T, que a su vez producen Recomendaciones sobre dichos temas.

La aprobación de Recomendaciones por los Miembros del UIT-T es el objeto del procedimiento establecido en la Resolución N.º 1 de la CMNT (Helsinki, 1 al 12 de marzo de 1993).

La Recomendación UIT-T G.729 ha sido preparada por la Comisión de Estudio 15 (1993-1996) del UIT-T y fue aprobada por el procedimiento de la Resolución N.º 1 de la CMNT el 19 de marzo de 1996.

NOTA

En esta Recomendación, la expresión «Administración» se utiliza para designar, en forma abreviada, tanto una administración de telecomunicaciones como una empresa de explotación reconocida de telecomunicaciones.

© UIT 1996

Es propiedad. Ninguna parte de esta publicación puede reproducirse o utilizarse, de ninguna forma o por ningún medio, sea éste electrónico o mecánico, de fotocopia o de microfilm, sin previa autorización escrita por parte de la UIT.

ÍNDICE

	<i>Página</i>
1	Introducción 1
2	Descripción general del codificador/decodificador (códec) 1
2.1	Codificador 2
2.2	Decodificador 3
2.3	Retardo..... 4
2.4	Descripción del códec de señales vocales..... 4
2.5	Convenciones de notación 4
3	Descripción de las funciones del codificador..... 7
3.1	Preprocesamiento..... 7
3.2	Análisis y cuantificación de la predicción lineal 7
3.3	Ponderación perceptual..... 14
3.4	Análisis de tono en bucle abierto..... 15
3.5	Cálculo de la respuesta de impulso..... 16
3.6	Cálculo de la señal objetivo 16
3.7	Búsqueda de la tabla de códigos adaptativos..... 17
3.8	Tabla de códigos fijos: estructura y búsqueda 19
3.9	Cuantificación de las ganancias..... 22
3.10	Actualización de la memoria 24
4	Descripción de las funciones del decodificador 25
4.1	Procedimiento de decodificación de los parámetros..... 25
4.2	Postprocesamiento 28
4.3	Inicialización del codificador y el decodificador..... 30
4.4	Ocultamiento de borrados de tramas..... 30
5	Descripción binaria exacta del códec CS-ACELP 32
5.1	Empleo del soporte lógico de simulación 32
5.2	Organización del soporte lógico de simulación 32

CODIFICACIÓN DE LA VOZ A 8 kbit/s MEDIANTE PREDICCIÓN LINEAL CON EXCITACIÓN POR CÓDIGO ALGEBRAICO DE ESTRUCTURA CONJUGADA

(Ginebra, 1996)

1 Introducción

En la presente Recomendación se describe un algoritmo para la codificación de la voz a 8 kbit/s mediante predicción lineal con excitación por código algebraico con estructura conjugada (CS-ACELP, *conjugate-structure algebraic-code-excited linear-prediction*).

El códec en cuestión está diseñado para operar con una señal digital obtenida tras efectuar, primero un filtrado con la anchura de banda telefónica (Recomendación G.712) de la señal analógica de entrada, seguido de su muestreo a 8000 Hz y su conversión a una modulación por impulsos codificados (MIC) lineal de 16 bits, para entrar en el codificador. La salida del decodificador deberá reconvertirse a una señal analógica siguiendo un método similar. Otras características de entrada/salida, como las que se especifican en la Recomendación G.711 para datos MIC de 64 kbit/s, deberán convertirse a MIC lineal de 16 bits antes de codificar, o de MIC lineal de 16 bits al formato apropiado después de decodificar. El tren de bits del codificador al decodificador se define dentro de esta norma.

La Recomendación está estructurada como sigue: la cláusula 2 presenta un resumen general del algoritmo CS-ACELP. En las cláusulas 3 y 4 se exponen, respectivamente, los principios del codificador y del decodificador CS-ACELP. La cláusula 5 describe el soporte lógico que define dicho códec en una aritmética de coma fija de 16 bits.

2 Descripción general del codificador/decodificador (códec)

El códec CS-ACELP se basa en el modelo de codificación mediante la predicción lineal con excitación por código (CELP). Opera con tramas vocales de 10 ms correspondientes a 80 muestras a una velocidad de muestreo de 8000 muestras por segundo. En cada trama de 10 ms se analiza la señal vocal para extraer los parámetros del modelo CELP (coeficientes de filtros de predicción lineal, ganancias e índices de las tablas de códigos adaptativos y fijos). Los parámetros en cuestión se codifican y se transmiten. El Cuadro 1 ilustra la asignación de bits para los parámetros del códec. En el decodificador, dichos parámetros se utilizan para recuperar los parámetros de excitación y del filtro de síntesis. La voz se reconstruye filtrando la excitación a través del filtro de síntesis de corto plazo, como se ve en la Figura 1. El filtro de síntesis de corto plazo se basa en un filtro de predicción lineal (PL) de décimo orden. El filtro de síntesis de largo plazo o de tono se aplica mediante el método de la llamada tabla de códigos adaptativos. Tras calcular la señal vocal reconstruida, ésta se mejora con un postfiltrado.

CUADRO 1/G.729

Asignación de bit de algoritmos CS-ACELP a 8 kbit/s por segundo (trama de 10 ms)

Parámetro	Palabra de código	Subtrama 1	Subtrama 2	Total por trama
Pares del espectro lineal (LSP)	$L0, L1, L2, L3$			18
Retardo de la tabla de códigos adaptativos	$P1, P2$	8	5	13
Paridad del retardo de tono	$P0$	1		1
Índice de tabla de códigos fijos	$C1, C2$	13	13	26
Signo de tabla de códigos fijos	$S1, S2$	4	4	8
Ganancias de tabla de códigos (fase 1)	$GA1, GA2$	3	3	6
Ganancias de tabla de códigos (fase 2)	$GB1, GB2$	4	4	8
Total				80

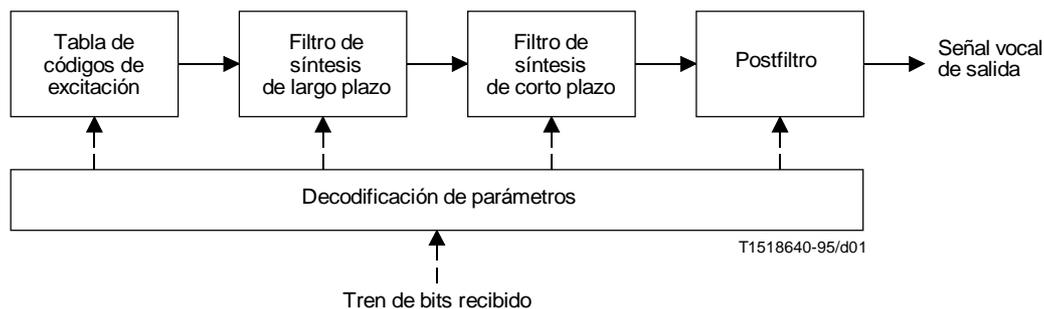


FIGURA 1/G.729

Diagrama funcional del modelo conceptual de síntesis (CELP)

2.1 Codificador

El principio de codificación puede observarse en la Figura 2. La señal de entrada pasa por un filtro de paso alto y se pone a escala en el bloque de preprocesamiento. La señal preprocesada actúa como señal de entrada para todo el análisis ulterior. Se efectúa un análisis de predicción lineal (LP) para cada trama de 10 ms con el fin de calcular los coeficientes de filtro LP. Éstos se convierten en pares del espectro lineal (LSP, *line spectrum pairs*), cuantificándose mediante una cuantificación vectorial (VQ) predictiva en dos etapas de 18 bits. La señal de excitación se selecciona utilizando un procedimiento de búsqueda basado en el análisis por síntesis, según el cual la diferencia entre la señal original y la reconstruida se reduce al mínimo de acuerdo con una medida de la distorsión ponderada perceptualmente. Esto se logra pasando la señal de error por un filtro de ponderación perceptual, cuyos coeficientes se derivan del filtro LP sin cuantificar. El valor de la ponderación perceptual se hace adaptativa, con el fin de mejorar la calidad para señales de entrada con una respuesta de frecuencia plana.

Los parámetros de excitación (parámetros de tabla de códigos fijos y adaptativos) se determinan para cada subtrama de 5 ms (40 muestras). Los coeficientes cuantificados y no cuantificados del filtro LP se aplican a la segunda subtrama, mientras que para la primera subtrama se utilizan coeficientes del filtro LP interpolados (cuantificados o no). Se estima un retardo de tono en bucle abierto por cada trama de 10 ms, en base a señal vocal ponderada perceptualmente. Luego se efectúan, para cada subtrama por separado, las siguientes operaciones. Se calcula la señal objetivo $x(n)$ pasando el LP residual por el filtro de síntesis ponderado $W(z)/\hat{A}(z)$. Los estados iniciales de estos filtros se actualizan filtrando la diferencia que se produce entre el residuo LP y la excitación. Ello equivale al método corriente de sustraer de la señal vocal ponderada la respuesta de entrada cero del filtro de síntesis ponderado. Se calcula la respuesta de impulso $h(n)$ del filtro de síntesis ponderado. Seguidamente se analiza el tono en bucle cerrado (para determinar el retardo y ganancia de la tabla de códigos adaptativos) mediante la respuesta objetivo $x(n)$ y la respuesta a los impulsos $h(n)$, indagando en torno al valor del retardo de tono de bucle abierto. Se utiliza un retardo de tono fraccionario de 1/3 de definición. El retardo de tono se codifica con 8 bits para la primera subtrama y diferencialmente con 5 bits para la segunda. La señal objetivo $x(n)$ se actualiza sustrayendo la contribución (filtrada) de la tabla de códigos adaptativos y se aplica este nuevo objetivo, $x'(n)$, para la búsqueda de la tabla de códigos fijos, con el fin de obtener la excitación óptima. Para la excitación de la tabla de códigos fijos se aplica una tabla de códigos algebraicos de 17 bits. Las ganancias de las contribuciones de las tablas de códigos adaptativos y fijos se cuantifican vectorialmente con 7 bits (con una predicción de media móvil aplicada a la ganancia de la tabla de códigos fijos). Finalmente, se actualizan las memorias de los filtros mediante la señal de excitación así determinada.

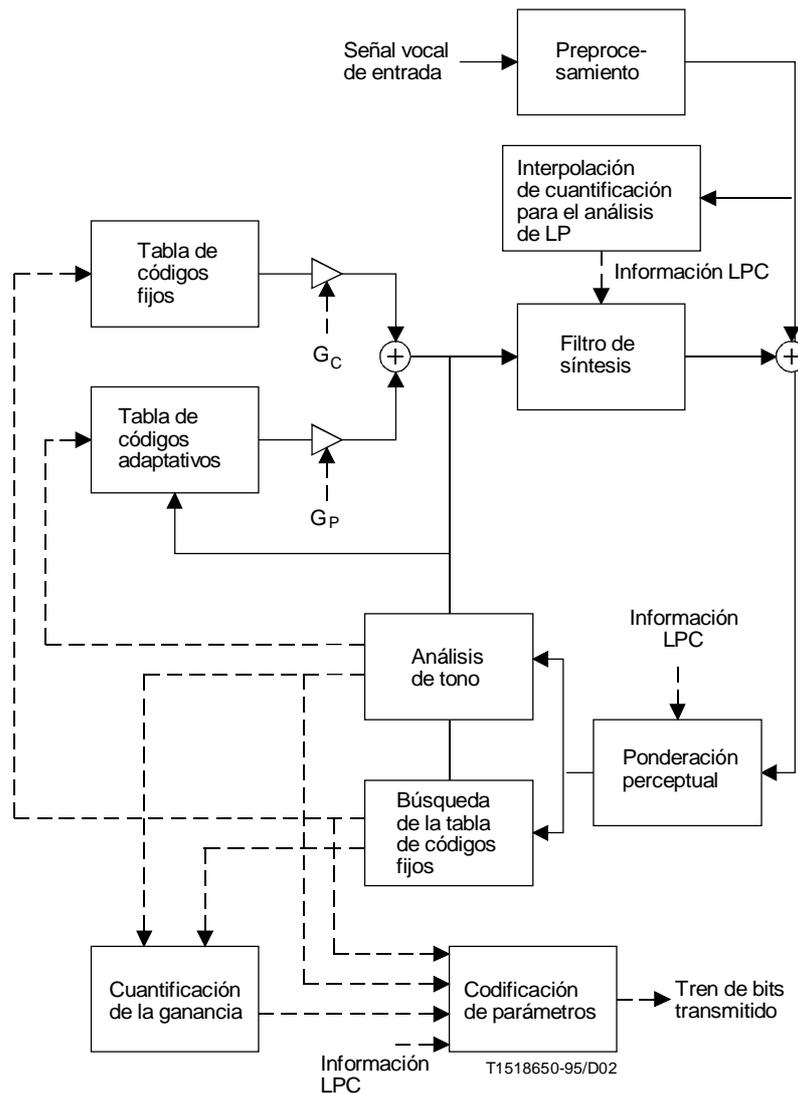


FIGURA 2/G.729

Principio de codificación del codificador CS-ACELP

2.2 Decodificador

El principio del decodificador aparece en la Figura 3. Primero se extraen los índices de los parámetros a partir del tren de bits recibido. Los índices se decodifican para obtener los parámetros del códec correspondientes a una trama de voz de 10 ms. Estos parámetros son los coeficientes LSP, los dos retardos de tono fraccionarios, los dos vectores de la tabla de códigos fijos y ambos conjuntos de ganancias de las tablas de códigos adaptativos y fijos. Los coeficientes LSP se interpolan y se convierten en coeficientes del filtro LP de cada subtrama. A continuación, para cada subtrama de 5 ms se aplican los siguientes pasos:

- se construye la excitación sumando los vectores de las tablas de los códigos adaptativos y fijos, puestos a escala por sus respectivas ganancias;
- se reconstruye la señal vocal filtrando la excitación por el filtro de síntesis LP;
- se hace pasar la señal vocal reconstruida a través de una fase de postprocesamiento, que incluye un postfiltro adaptativo basado en filtros de síntesis de largo y corto plazo, seguido de un filtro de paso alto y un escalamiento.

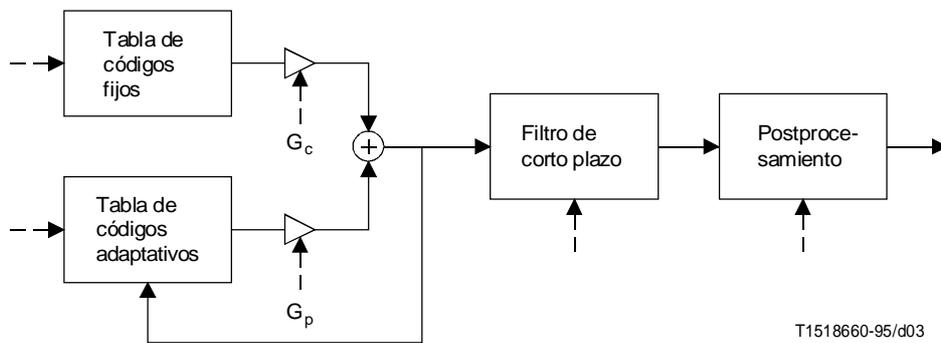


FIGURA 3/G.729
Principio del decodificador CS-ACELP

2.3 Retardo

El códec codifica la voz y otras señales de audio con tramas de 10 ms. Se produce además un preanálisis de 5 ms, por lo que el retardo algorítmico total es de 15 ms. Los demás retardos producidos por la aplicación práctica de este códec tienen por causa:

- el tiempo de procesamiento necesario para las operaciones de codificación y decodificación;
- el tiempo de transmisión en el enlace de comunicación;
- el retardo de multiplexación por la combinación de datos de señales vocales y otros.

2.4 Descripción del códec de señales vocales

La descripción del algoritmo de codificación/decodificación de la voz en la presente Recomendación se hace en términos de operaciones matemáticas de coma fija y exactitud de bits. El código C de ANSI, indicado en la cláusula 5 y que es parte integrante de esta Recomendación, refleja este método descriptivo de coma fija y exactitud de bit. Las descripciones matemáticas del codificador (véase la cláusula 3) y del decodificador (véase la cláusula 4) pueden aplicarse también de varias otras maneras, dando lugar, quizás, a aplicaciones del códec que no satisfacen los términos de esta Recomendación. Por este motivo, la descripción del algoritmo del código C de ANSI que figura en la cláusula 5 prevalecerá en caso de discrepancia con cualquier otra descripción matemática contenida en las cláusulas 3 y 4. Sin llegar a ser exhaustivo, puede obtenerse de la UIT un juego de señales de prueba utilizables al aplicar el código C de ANSI.

2.5 Convenciones de notación

En todo este documento se ha procurado seguir las siguientes convenciones de notación:

- Las tablas de códigos se destacan por medio de caracteres caligráficos (por ejemplo \mathcal{C}).
- Las señales de tiempo se destacan mediante su símbolo seguido de un índice de muestra entre paréntesis [por ejemplo $s(n)$]. El símbolo n se utiliza como índice de muestra.
- Los superíndices entre paréntesis (por ejemplo $g^{(m)}$) se utilizan para indicar la dependencia en el tiempo de las variables. La variable m corresponde, según el contexto, a un índice de trama o de subtrama, mientras que la variable n corresponde a un índice de muestra.
- Los índices de recursión se identifican mediante un superíndice entre corchetes (por ejemplo $E^{[k]}$).
- Los subíndices identifican un elemento particular de una matriz de coeficientes.
- El símbolo $\hat{}$ identifica la versión cuantificada de un parámetro (por ejemplo \hat{g}_c).
- Entre corchetes se expresan los intervalos de los parámetros, incluyendo sus límites (por ejemplo $[0,6, 0,9]$).

- La función *log* indica un logaritmo decimal.
- La función *int* indica truncamiento a sus valores enteros.
- Los números decimales con coma flotante utilizados son versiones redondeadas de los valores utilizados en la aplicación del código C de ANSI con coma fija a 16 bits.

En el Cuadro 2 se señalan los símbolos más importantes utilizados en esta Recomendación. Un glosario de las señales más importante figura en el Cuadro 3. El Cuadro 4 contiene un resumen de las variables del caso y sus dimensiones. Los parámetros constantes aparecen en el Cuadro 5. Un resumen de las abreviaturas utilizadas en esta Recomendación figura en el Cuadro 6.

CUADRO 2/G.729

Glosario de los símbolos más importantes

Denominación	Referencia	Designación
$1/\hat{A}(z)$	Ecuación (2)	Filtro de síntesis LP
$H_{h1}(z)$	Ecuación (1)	Filtro paso alto de entrada
$H_p(z)$	Ecuación (78)	Postfiltro de largo plazo
$H_f(z)$	Ecuación (84)	Postfiltro de corto plazo
$H_t(z)$	Ecuación (86)	Filtro de compensación de pendiente
$H_{h2}(z)$	Ecuación (91)	Filtro paso alto de salida
$P(z)$	Ecuación (46)	Prefiltro para tabla de códigos fijos
$W(z)$	Ecuación (27)	Filtro de ponderación

CUADRO 3/G.729

Glosario de las señales más importantes

Denominación	Referencia	Designación
$c(n)$	3.8	Contribución de la tabla de códigos fijos
$d(n)$	3.8.1	Correlación entre la señal objetivo y $h(n)$
$ew(n)$	3.10	Señal de error
$h(n)$	3.5	Respuesta de impulso de los filtros de ponderación y síntesis
$r(n)$	3.6	Señal residual
$s(n)$	3.1	Señal vocal preprocesada
$\hat{s}(n)$	4.1.6	Señal vocal reconstruida
$s'(n)$	3.2.1	Señal vocal ventanizada
$sf(n)$	4.2	Salida del postfiltro
$sf'(n)$	4.2	Salida del postfiltro con escalamiento de ganancia
$sw(n)$	3.6	Señal vocal ponderada
$x(n)$	3.6	Señal objetivo
$x'(n)$	3.8.1	Segunda señal objetivo
$u(n)$	3.10	Excitación del filtro de síntesis LP
$v(n)$	3.7.1	Contribución de tabla de códigos adaptativos
$y(n)$	3.7.3	Convolución $v(n) * h(n)$
$z(n)$	3.9	Convolución $c(n) * h(n)$

CUADRO 4/G.729

Glosario de las variables más importantes

Denominación	Tamaño	Designación
g_p	1	Ganancia de tabla de códigos adaptativos
g_c	1	Ganancia de tabla de códigos fijos
g_l	1	Término ganancia para el postfiltro de largo plazo
g_f	1	Término ganancia para el postfiltro de corto plazo
g_t	1	Término ganancia para el postfiltro de pendiente
G	1	Ganancia para la normalización de ganancia
T_{op}	1	Retardo de tono en bucle abierto
a_i	11	Coefficientes LP ($a_0 = 1,0$)
k_i	10	Coefficientes de reflexión
k'_1	1	Coefficiente de reflexión para el postfiltro de pendiente
o_i	2	Coefficientes LAR
ω_i	10	Frecuencias normalizadas LSF
$\hat{p}_{i,j}$	40	Predictor de media móvil para cuantificación LSF
q_i	10	Coefficientes LSP
$r(k)$	11	Coefficientes de autocorrelación
$r'(k)$	11	Coefficientes de autocorrelación modificados
w_i	10	Coefficientes de ponderación LSP
\hat{l}_i	10	Salida de cuantificador LSP

CUADRO 5/G.729

Glosario de las constantes más importantes

Denominación	Valor	Designación
f_s	8000	Frecuencia de muestreo
f_0	60	Expansión de anchura de banda
γ_1	0,94/0,98	Factor de ponderación del filtro de ponderación perceptual
γ_2	0,60/[0,4 – 0,7]	Factor de ponderación del filtro de ponderación perceptual
γ_n	0,55	Factor de ponderación del postfiltro
γ_d	0,70	Factor de ponderación del postfiltro
γ_p	0,50	Factor de ponderación del postfiltro de tono
γ_t	0,90/0,2	Factor de ponderación del postfiltro de pendiente
\mathcal{C}	Cuadro 7	Tabla de códigos fijos (algebraicos)
L0	3.2.4	Tabla de códigos del predictor de media móvil
L1	3.2.4	Tabla de códigos LSP, primera etapa
L2	3.2.4	Tabla de códigos LSP, segunda etapa (parte inferior)
L3	3.2.4	Tabla de códigos LSP, segunda etapa (parte superior)
\mathcal{G}_A	3.9	Tabla de códigos de ganancias (primera etapa)
\mathcal{G}_B	3.9	Tabla de códigos de ganancias (segunda etapa)
w_{lag}	Ecuación (6)	Ventana de retardo de correlación
w_{lp}	Ecuación (3)	Ventana de análisis LP

Glosario de abreviaturas

Abreviatura	Designación
CELP	Predicción lineal con excitación por código (<i>code-excited linear-prediction</i>)
CS-ACELP	CELP con código algebraico de estructura conjugada (<i>conjugate-structure algebraic-CELP</i>)
MA	Media móvil (<i>moving average</i>)
MSB	Bit más significativo (<i>most significant bit</i>)
MSE	Error cuadrático medio (<i>mean-squared error</i>)
LAR	Logaritmo de relación de área (<i>log area ratio</i>)
LP	Predicción lineal (<i>linear prediction</i>)
LSP	Par del espectro lineal (<i>line spectral pair</i>)
LSF	Frecuencia del espectro lineal (<i>line spectral frequency</i>)
VQ	Cuantificación vectorial (<i>vector quantization</i>)

3 Descripción de las funciones del codificador

En esta cláusula se describen las diferentes funciones del codificador representadas por los bloques de la Figura 2. Un flujo de las señales más detallado aparece en la Figura 4.

3.1 Preprocesamiento

Como se indica en la cláusula 2, se supone que la entrada del codificador vocal es una señal MIC (modulación por impulsos codificados) de 16 bits. Dos funciones de preprocesamiento tienen lugar antes de iniciarse la codificación:

- 1) el escalamiento de las señales, y
- 2) el filtrado de paso alto.

Para poner en escala se divide la entrada por 2, con el fin de reducir la posibilidad de desbordamientos en la aplicación de coma fija. El filtro de paso alto es una precaución para evitar componentes indeseables de baja frecuencia. Se utiliza un filtro de polos y ceros de segundo orden con una frecuencia de corte de 140 Hz. Ambas funciones, el escalamiento y el filtrado de paso alto, se combinan dividiendo los coeficientes del numerador de este filtro por 2. El filtro resultante está dado por la expresión:

$$H_{h1}(z) = \frac{0,46363718 - 0,92724705z^{-1} + 0,46363718z^{-2}}{1 - 1,9059465z^{-1} + 0,9114024z^{-2}} \quad (1)$$

La señal de entrada filtrada a través de $H_{h1}(z)$ se define como $s(n)$, símbolo que se utilizará en todas las demás operaciones del códec.

3.2 Análisis y cuantificación de la predicción lineal

Los filtros de análisis y síntesis de corto plazo se basan en filtros de predicción lineal (LP) de 10.º orden. El filtro de síntesis LP se define por:

$$\frac{1}{\hat{A}(z)} = \frac{1}{1 + \sum_{i=1}^{10} \hat{a}_i z^{-i}} \quad (2)$$

donde \hat{a}_i , $i = 1, \dots, 10$, representa los coeficientes (cuantificados) de predicción lineal (LP). La predicción a corto plazo o análisis de predicción lineal se lleva a cabo para cada trama vocal mediante el método de autocorrelación, con una ventana asimétrica de 30 ms. Cada 80 muestras (10 ms), se calculan los coeficientes de autocorrelación de las señales vocales ventanizadas y se convierten en coeficientes LP por medio del algoritmo de Levinson. Luego los coeficientes LP se transforman al dominio LSP, para permitir la cuantificación e interpolación. Los filtros interpolados, cuantificados o no, se reconvierten en coeficientes de filtro LP (a fin de construir los filtros de síntesis y ponderación para cada subtrama).

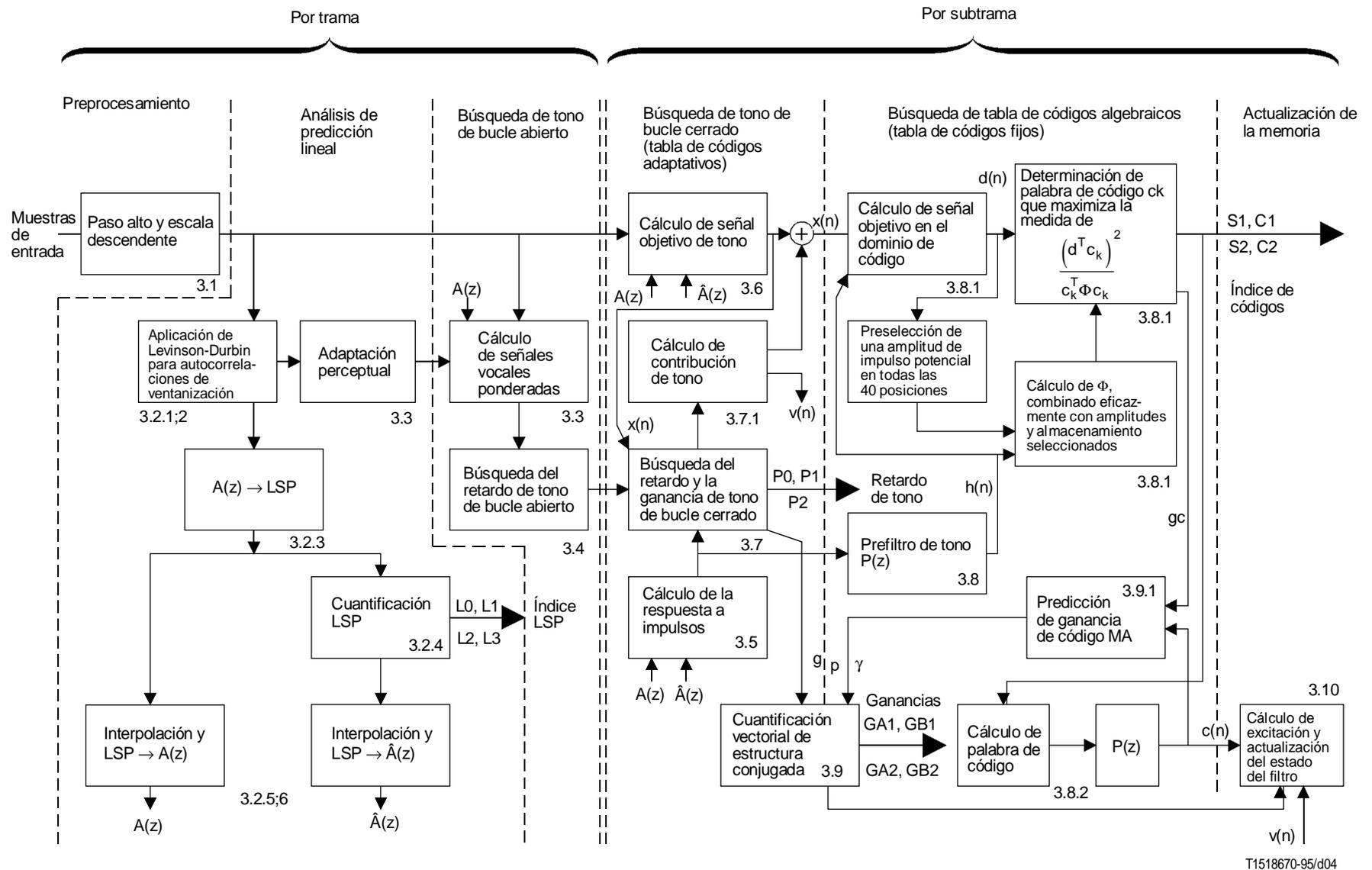


FIGURA 4/G.729

Flujo de señales en el codificador CS-ACELP

3.2.1 Ventanización y cálculo de la autocorrelación

La ventana de análisis LP consta de dos partes: la primera es media ventana de Hamming y la segunda un cuarto del ciclo de la función coseno. La ventana está dada por la ecuación:

$$w_{lp}(n) = \begin{cases} 0,54 - 0,46 \cos\left(\frac{2\pi n}{399}\right) & n = 0, \dots, 199 \\ \cos\left(\frac{2\pi(n - 200)}{159}\right) & n = 200, \dots, 239 \end{cases} \quad (3)$$

Como en el análisis de predicción lineal hay un preanálisis de 5 ms, se requieren 40 muestras de la trama vocal futura. Esto se traduce en un retardo algorítmico suplementario de 5 ms en la fase del codificador. La ventana de análisis LP se aplica a 120 muestras de las tramas vocales precedentes, 80 muestras de la trama vocal presente y 40 muestras de la trama futura. El procedimiento de ventanización se ilustra gráficamente en la Figura 5.

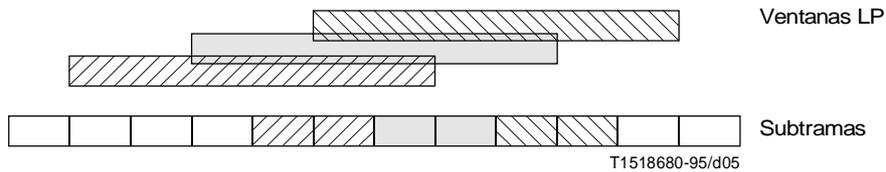


FIGURA 5/G.729

Procedimiento de ventanización en el análisis LP

El diferente sombreado identifica las correspondientes excitación y ventanas de análisis LP.

La señal vocal ventanizada:

$$s'(n) = w_{lp}(n) s(n) \quad n = 0, \dots, 239 \quad (4)$$

se utiliza para calcular los coeficientes de autocorrelación:

$$r(k) = \sum_{n=k}^{239} s'(n) s'(n - k) \quad k = 0, \dots, 10 \quad (5)$$

Con el fin de eludir problemas aritméticos para las señales de entrada de bajo nivel, el valor $r(0)$ tiene un límite inferior de $r(0) = 1,0$. Se produce una expansión de la anchura de banda de 60 Hz multiplicando los coeficientes de autocorrelación por:

$$w_{lag}(k) = \exp\left[-\frac{1}{2} \left(\frac{2\pi f_0 k}{f_s}\right)^2\right] \quad k = 1, \dots, 10 \quad (6)$$

donde $f_0 = 60$ Hz es la expansión de la anchura de banda y $f_s = 8000$ Hz es la frecuencia de muestreo. Se multiplica además $r(0)$ por un factor de corrección por ruido blanco 1,0001, lo que equivale a añadir un umbral de ruido de -40 dB. Los coeficientes de autocorrelación modificados están definidos por:

$$\begin{aligned} r'(0) &= 1,0001 r(0) \\ r'(k) &= w_{lag}(k) r(k) \quad k = 1, \dots, 10 \end{aligned} \quad (7)$$

3.2.2 Algoritmo de Levinson-Durbin

Los coeficientes de autocorrelación modificados $r'(k)$ se utilizan para obtener los coeficientes de filtro LP a_i , $i = 1, \dots, 10$, resolviendo el conjunto de ecuaciones:

$$\sum_{i=1}^{10} a_i r'(|i - k|) = -r'(k), \quad k = 1, \dots, 10 \quad (8)$$

El conjunto de ecuaciones (8) se resuelve mediante el algoritmo de Levinson-Durbin, basado en la siguiente recursión:

$$\begin{aligned} E^{[0]} &= r'(0) \\ \text{para } i &= 1 \text{ hasta } 10 \\ a_0^{[i-1]} &= 1 \\ k_i &= - \left[\sum_{j=0}^{i-1} a_j^{[i-1]} r'(i-j) \right] / E^{[i-1]} \\ a_i^{[i]} &= k_i \\ \text{para } j &= 1 \text{ hasta } i - 1 \\ a_j^{[i]} &= a_j^{[i-1]} + k_i a_{i-j}^{[i-1]} \\ \text{fin} \\ E^{[i]} &= (1 - k_i^2) E^{[i-1]} \\ \text{fin} \end{aligned}$$

La solución final se expresa por $a_j = a_j^{[10]}$, $j = 0, \dots, 10$, con $a_0 = 1,0$.

3.2.3 Conversión de LP a LSP

Los coeficientes de filtro LP a_i , $i = 0, \dots, 10$ se convierten para los efectos de cuantificación e interpolación en coeficientes de pares del espectro lineal (LSP). Para un filtro LP de 10.º orden se definen los coeficientes LSP como las raíces de los polinomios suma y diferencia:

$$F_1'(z) = A(z) + z^{-11}A(z^{-1}) \quad (9)$$

y

$$F_2'(z) = A(z) - z^{-11}A(z^{-1}) \quad (10)$$

respectivamente. El polinomio $F_1'(z)$ es simétrico y el $F_2'(z)$ es antisimétrico. Puede demostrarse que todas las raíces de estos polinomios se encuentran en el círculo unidad y alternan entre sí. $F_1'(z)$ tiene una raíz $z = -1$ ($\omega = \pi$), mientras que $F_2'(z)$ tiene una raíz $z = 1$ ($\omega = 0$). Ambas raíces se eliminan al definirse los nuevos polinomios:

$$F_1(z) = F_1'(z) / (1 + z^{-1}) \quad (11)$$

y

$$F_2(z) = F_2'(z) / (1 - z^{-1}) \quad (12)$$

Cada uno de estos polinomios tiene cinco raíces conjugadas en el círculo unidad ($e^{\pm j\omega_i}$), que pueden expresarse como:

$$F_1(z) = \prod_{i=1,3,\dots,9} (1 - 2q_i z^{-1} + z^{-2}) \quad (13)$$

y

$$F_2(z) = \prod_{i=2,4,\dots,10} (1 - 2q_i z^{-1} + z^{-2}) \quad (14)$$

donde $q_i = \cos(\omega_i)$. Los coeficientes ω_i son las frecuencias (LSF, *line spectral frequencies*) y satisfacen la propiedad de ordenación $0 < \omega_1 < \omega_2 < \dots < \omega_{10} < \pi$. Los coeficientes q_i se definen como coeficientes LSP en el dominio coseno.

Puesto que los polinomios $F_1(z)$ y $F_2(z)$ son simétricos, sólo es necesario calcular los primeros cinco coeficientes de cada uno. Los coeficientes de estos polinomios se obtienen mediante las relaciones recursivas:

$$\begin{aligned} f_1(i+1) &= a_{i+1} + a_{10-i} - f_1(i) & i &= 0, \dots, 4 \\ f_2(i+1) &= a_{i+1} - a_{10-i} + f_2(i) & i &= 0, \dots, 4 \end{aligned} \quad (15)$$

donde $f_1(0) = f_2(0) = 1,0$. Los coeficientes LSP se obtienen evaluando los polinomios $F_1(z)$ y $F_2(z)$ en 60 puntos igualmente espaciados entre 0 y π y controlando los cambios de signo. Todo cambio de signo implica la existencia de una raíz y el intervalo de cambio de signo se divide entonces cuatro veces, para permitir un mejor seguimiento de la raíz. Se utilizan los polinomios de Chebyshev para evaluar $F_1(z)$ y $F_2(z)$. Por este método, las raíces se obtienen directamente en el dominio coseno. Los polinomios $F_1(z)$ o $F_2(z)$, evaluados en $z = e^{j\omega}$, pueden expresarse como:

$$F(\omega) = 2e^{-j5\omega} C(x) \quad (16)$$

siendo

$$C(x) = T_5(x) + f(1)T_4(x) + f(2)T_3(x) + f(3)T_2(x) + f(4)T_1(x) + f(5)/2 \quad (17)$$

donde $T_m(x) = \cos(m\omega)$ es el polinomio de Chebyshev de m-ésimo orden y $f(i)$, $i = 1, \dots, 5$, representa los coeficientes de $F_1(z)$ o $F_2(z)$, calculados mediante la ecuación (15). El polinomio $C(x)$ se evalúa para un determinado valor de $x = \cos(\omega)$ mediante la relación recursiva:

para $k = de\ 4\ a\ 1$

$$b_k = 2xb_{k+1} - b_{k+2} + f(5 - k)$$

fin

$$C(x) = xb_1 - b_2 + f(5)/2$$

siendo los valores iniciales $b_5 = 1$ y $b_6 = 0$.

3.2.4 Cuantificación de los coeficientes LSP

Los coeficientes LSP q_i se cuantifican mediante la representación ω_i de LSF en el dominio normalizado de la frecuencia $[0, \pi]$, o sea:

$$\omega_i = \arccos(q_i) \quad i = 1, \dots, 10 \quad (18)$$

Se aplica una predicción MA de 4.º orden conmutada para predecir los coeficientes LSF de la trama en curso. La diferencia entre los coeficientes calculados y su predicción se cuantifica mediante un cuantificador vectorial de dos fases. La primera fase es una cuantificación vectorial (VQ) de 10 dimensiones que utiliza una tabla de códigos L1 de 128 entradas (7 bits). La segunda fase es una VQ de 10 bits que se aplica como una VQ de división mediante dos tablas de códigos de cinco dimensiones, L2 y L3, de 32 entradas (5 bits) cada una.

Con el fin de explicar el proceso de cuantificación, es conveniente primero describir el proceso de decodificación. Cada coeficiente se obtiene de la suma de dos tablas de códigos:

$$\hat{l}_i = \begin{cases} L1_i(L1) + L2_i(L2) & i = 1, \dots, 5 \\ L1_i(L1) + L3_{i-5}(L3) & i = 6, \dots, 10 \end{cases} \quad (19)$$

donde $L1$, $L2$ y $L3$ son índices de tabla de códigos. Para evitar resonancias bruscas en el filtro de síntesis cuantificado LP, los coeficientes \hat{l}_i se ordenan de modo que los coeficientes adyacentes estén a una distancia mínima de J . La secuencia de reordenamiento se presenta a continuación:

para $i = 2, \dots, 10$

si $(\hat{l}_{i-1} > \hat{l}_i - J)$

$$\hat{l}_{i-1} = (\hat{l}_i + \hat{l}_{i-1} - J)/2$$

$$\hat{l}_i = (\hat{l}_i + \hat{l}_{i-1} + J)/2$$

fin

fin

Este proceso de reordenamiento se hace dos veces. La primera vez se establece un valor de $J = 0,0012$, y la segunda vez de $J = 0,0006$. Después de efectuado el reordenamiento, los coeficientes LSF cuantificados $\hat{\omega}_i^{(m)}$ para la trama presente m se obtienen de la suma ponderada de las salidas de cuantificador precedentes $\hat{l}_i^{(m-k)}$ y de la salida actual del cuantificador $\hat{l}_i^{(m)}$:

$$\hat{\omega}_i^{(m)} = \left(1 - \sum_{k=1}^4 \hat{p}_{i,k} \right) \hat{l}_i^{(m)} + \sum_{k=1}^4 \hat{p}_{i,k} \hat{l}_i^{(m-k)} \quad i = 1, \dots, 10 \quad (20)$$

donde $\hat{p}_{i,k}$ son los coeficientes del predictor MA conmutado. Para definir qué predictor MA ha de utilizarse se recurre a un bit $L0$ separado. En el arranque, los valores iniciales de $\hat{l}_i^{(k)}$ están dados por $\hat{l}_i = i\pi/11$ para todo $k < 0$.

Una vez calculado $\hat{\omega}_i$, se controla la estabilidad del filtro correspondiente. El procedimiento es el siguiente:

- 1) se ordenan los valores del coeficiente $\hat{\omega}_i$ de menor a mayor;
- 2) si $\hat{\omega}_i < 0,005$ entonces $\hat{\omega}_i = 0,005$;
- 3) si $\hat{\omega}_{i+1} - \hat{\omega}_i > 0,0391$, entonces $\hat{\omega}_{i+1} = \hat{\omega}_i + 0,0391$, $i = 1, \dots, 9$;
- 4) si $\hat{\omega}_{10} > 3,135$ entonces $\hat{\omega}_{10} = 3,135$.

El procedimiento para codificar los parámetros LSF puede resumirse como sigue. Se procurará hallar para cada uno de los dos predictores MA la mejor aproximación a los coeficientes LSF del momento. Como mejor aproximación se considera la que minimiza el error cuadrático medio ponderado:

$$E_{lsf} = \sum_{i=1}^{10} w_i (\omega_i - \hat{\omega}_i)^2 \quad (21)$$

Los pesos w_i se hacen adaptativos como función de los coeficientes LSF no cuantificados,

$$w_i = \begin{cases} 1,0 & \text{si } \omega_2 - 0,04\pi - 1 > 0 \\ 10 (\omega_2 - 0,04\pi - 1)^2 + 1 & \text{en los demás casos} \end{cases}$$

$$w_i \quad 2 \leq i \leq 9 = \begin{cases} 1,0 & \text{si } \omega_{i+1} - \omega_{i-1} - 1 > 0 \\ 10 (\omega_{i+1} - \omega_{i-1} - 1)^2 + 1 & \text{en los demás casos} \end{cases} \quad (22)$$

$$w_{10} = \begin{cases} 1,0 & \text{si } -\omega_9 + 0,92\pi - 1 > 0 \\ 10 (-\omega_9 + 0,92\pi - 1)^2 + 1 & \text{en los demás casos} \end{cases}$$

Además, los pesos w_5 y w_6 se multiplican cada uno por 1,2.

El vector a cuantificar para la trama actual m se obtiene mediante:

$$l_i = \left[\omega_i^{(m)} - \sum_{k=1}^4 \hat{p}_{i,k} \hat{l}_i^{(m-k)} \right] / \left(1 - \sum_{k=1}^4 \hat{p}_{i,k} \right) \quad i = 1, \dots, 10 \quad (23)$$

Se indaga la primera tabla de códigos L1, seleccionando la entrada L1 que hace mínimo el error cuadrático medio (no ponderado). Seguidamente se indaga la segunda tabla de códigos L2, que define la parte inferior de la segunda fase. Para cada posible candidato, se reconstruye el vector parcial $\hat{\omega}_i$, $i = 1, \dots, 5$, mediante la ecuación (20), reordenándolo para asegurar una distancia mínima de 0,0012. Se calcula el error cuadrático medio ponderado según la ecuación (21) y se selecciona el vector L2 que produzca el error más bajo. Aplicando el vector L1 seleccionado de la primera fase y la parte inferior del L2 de la segunda fase, se extrae la parte superior de la segunda fase de la tabla de códigos L3. Se procede a reordenar nuevamente los valores para asegurar una distancia mínima de 0,0012. Se selecciona el vector L3 que minimice el error cuadrático medio ponderado. El vector resultante \hat{l}_i , $i = 1, \dots, 10$ se reordena para asegurar una distancia mínima de 0,0006. Este proceso se lleva a cabo para cada uno de los dos predictores de media móvil (MA) definidos por L0, seleccionando el predictor de MA L0 que produzca el menor error cuadrático medio ponderado. Como se indica al comienzo de esta cláusula, el reordenamiento del vector resultante \hat{l}_i se efectúa dos veces, procediéndose a un control de estabilidad para obtener los coeficientes LSF cuantificados $\hat{\omega}_i$.

3.2.5 Interpolación de los coeficientes LSP

Los coeficientes LP cuantificados (y no cuantificados) se utilizan para la segunda subtrama. Para la primera subtrama, los coeficientes LP cuantificados (y no cuantificados) se obtienen mediante interpolación lineal de los parámetros correspondientes en las subtramas adyacentes. La interpolación tiene lugar en los coeficientes LSP del dominio coseno. Sean $q_i^{(actual)}$ los coeficientes LSP calculados para la trama actual de 10 ms, y $q_i^{(anterior)}$ los coeficientes LSP calculados en la trama anterior de 10 ms. Los coeficientes LSP interpolados (no cuantificados) en cada una de las dos subtramas corresponden a:

$$\begin{aligned} \text{Subtrama 1: } q_i^{(1)} &= 0,5q_i^{(anterior)} + 0,5q_i^{(actual)} & i &= 1, \dots, 10 \\ \text{Subtrama 2: } q_i^{(2)} &= q_i^{(actual)} & i &= 1, \dots, 10 \end{aligned} \quad (24)$$

El mismo procedimiento de interpolación se aplica a los coeficientes LSP cuantificados, sustituyendo q_i por \hat{q}_i en la ecuación (24).

3.2.6 Conversión de LSP a LP

Una vez cuantificados e interpolados los coeficientes LSP, se reconvierten en coeficientes LP a_i . Esta conversión se realiza de la siguiente manera. Se hallan los coeficientes de $F_1(z)$ y $F_2(z)$ ampliando las ecuaciones (13) y (14) tras conocer los coeficientes LSP cuantificados e interpolados. Los coeficientes $f_1(i)$, $i = 1, \dots, 5$, se calculan a partir de q_i mediante la relación recursiva:

para $i = 1$ a 5

$$f_1(i) = -2q_{2i-1}f_1(i-1) + 2f_1(i-2)$$

para $j = i - 1$ decreciendo hasta 1

$$f_1^{[i]}(j) = f_1^{[i-1]}(j) - 2q_{2i-1}f_1^{[i-1]}(j-1) + f_1^{[i-1]}(j-2)$$

fin

fin

con valores iniciales $f_1(0) = 1$ y $f_1(-1) = 0$. Los coeficientes $f_2(i)$ se calculan de manera similar, sustituyendo q_{2i-1} por q_{2i} .

Después de determinar los coeficientes $f_1(i)$ y $f_2(i)$, $F_1(z)$ y $F_2(z)$ se multiplican por el factor $1 + z^{-1}$ y $1 - z^{-1}$ respectivamente, obteniendo $F_1'(z)$ y $F_2'(z)$, o sea:

$$\begin{aligned} f_1'(i) &= f_1(i) + f_1(i-1) & i &= 1, \dots, 5 \\ f_2'(i) &= f_2(i) - f_2(i-1) & i &= 1, \dots, 5 \end{aligned} \quad (25)$$

Finalmente, los coeficientes LP se calculan a partir de $f_1'(i)$ y $f_2'(i)$ mediante:

$$a_i = \begin{cases} 0,5f_1'(i) + 0,5f_2'(i) & i = 1, \dots, 5 \\ 0,5f_1'(11-i) - 0,5f_2'(11-i) & i = 6, \dots, 10 \end{cases} \quad (26)$$

Esto se deduce directamente de la relación $A(z) = (F_1'(z) + F_2'(z))/2$, así como del hecho de que $F_1'(z)$ y $F_2'(z)$ son, respectivamente, polinomios simétricos y antisimétricos.

3.3 Ponderación perceptual

El filtro de ponderación perceptual se basa en los coeficientes de filtro LP no cuantificados a_i y viene definido por:

$$W(z) = \frac{A(z/\gamma_1)}{A(z/\gamma_2)} = \frac{1 + \sum_{i=1}^{10} \gamma_1^i a_i z^{-i}}{1 + \sum_{i=1}^{10} \gamma_2^i a_i z^{-i}} \quad (27)$$

Los valores de γ_1 y γ_2 determinan la respuesta de frecuencia del filtro $W(z)$. Mediante un ajuste adecuado de estas variables es posible lograr una ponderación más eficaz. El método consiste en hacer de γ_1 y γ_2 una función de la forma espectral de la señal de entrada. Se efectúa esta adaptación una vez por trama de 10 ms, pero aplicando un procedimiento de interpolación para cada primera subtrama, a fin de suavizar el proceso de adaptación. La forma del espectro se obtiene a partir de un filtro de predicción lineal de 2.º orden, como resultado secundario de la recursión de Levinson-Durbin (véase 3.2.2). Los coeficientes de reflexión k_i se convierten en coeficientes o_i , logaritmo de relación de zona (LAR, *log area ratio*) mediante:

$$o_i = \log \frac{(1,0 + k_i)}{(1,0 - k_i)} \quad i = 1, 2 \quad (28)$$

Los coeficientes LAR correspondientes a la trama actual de 10 ms sirven para la segunda subtrama. Los coeficientes LAR de la primera subtrama se obtienen mediante interpolación lineal de los parámetros LAR de la trama anterior. Los coeficientes LAR interpolados en cada una de ambas subtramas están dados por:

$$\begin{aligned} \text{Subtrama 1: } o_i^{(1)} &= 0,5 o_i^{(\text{anterior})} + 0,5 o_i^{(\text{actual})} & i = 1, 2 \\ \text{Subtrama 2: } o_i^{(2)} &= o_i^{(\text{actual})} & i = 1, 2 \end{aligned} \quad (29)$$

La envolvente del espectro se caracteriza como plana ($flat = 1$) o inclinada ($flat = 0$). Para cada subtrama se logra esta caracterización aplicando a los coeficientes LAR una función umbral. Para evitar cambios bruscos, se realiza una histéresis tomando en cuenta el valor de $flat$ en la subtrama anterior $m - 1$,

$$flat^{(m)} = \begin{cases} 0 & \text{si } o_1^{(m)} < -1,74 \text{ y } o_2^{(m)} > 0,65 \text{ y } flat^{(m-1)} = 1 \\ 1 & \text{si } \left(o_1^{(m)} > -1,52 \text{ o bien } o_2^{(m)} < 0,43 \right) \text{ y } flat^{(m-1)} = 0 \\ flat^{(m-1)} & \text{en los otros casos} \end{cases} \quad (30)$$

Cuando el espectro interpolado para una subtrama se califica de plano ($flat^{(m)} = 1$), los factores de ponderación se establecen en $\gamma_1 = 0,94$ y $\gamma_2 = 0,6$. Cuando el espectro se califica de inclinado ($flat^{(m)} = 0$), el valor de γ_1 se establece a 0,98 y el de γ_2 se adapta a la intensidad de las resonancias en el filtro de síntesis LP, pero limitado entre 0,4 y 0,7. En caso de registrarse una fuerte resonancia, el valor de γ_2 se fija más cerca del límite superior. Esta adaptación se logra en base al criterio de la distancia mínima entre dos coeficientes LSP sucesivos para la subtrama actual. La distancia mínima está dada por:

$$d_{\text{mín}} = \text{mín} [\omega_{i+1} - \omega_i] \quad i = 1, \dots, 9 \quad (31)$$

El valor de γ_2 se calcula mediante la relación lineal:

$$\gamma_2 = -6,0 d_{\text{mín}} + 1,0 \quad \text{en el intervalo } 0,4 \leq \gamma_2 \leq 0,7 \quad (32)$$

La señal vocal ponderada en una subtrama está dada por:

$$sw(n) = s(n) + \sum_{i=1}^{10} a_i \gamma_1^i s(n - i) - \sum_{i=1}^{10} a_i \gamma_2^i sw(n - i) \quad n = 0, \dots, 39 \quad (33)$$

La señal vocal ponderada $sw(n)$ se utiliza para estimar el retardo de tono en la trama vocal.

3.4 Análisis de tono en bucle abierto

Con el fin de reducir la complejidad que implica buscar el mejor retardo de la tabla de códigos adaptativos, el campo de búsqueda se limita en torno a un retardo candidato T_{op} , que se obtiene de un análisis de tono en bucle abierto. Este análisis de tono en bucle abierto se efectúa una vez por cada trama (10 ms). La estimación de tono en bucle abierto utiliza los valores de señal vocal ponderada $sw(n)$ de la ecuación (33), y el método descrito a continuación: como primer paso, se establecen tres máximos de la correlación:

$$R(k) = \sum_{n=0}^{79} sw(n) sw(n - k) \quad (34)$$

para las tres gamas siguientes:

$$i = 1: 80, \dots, 143$$

$$i = 2: 40, \dots, 79$$

$$i = 3: 20, \dots, 39$$

Los máximos retenidos $R(t_i)$, $i = 1, \dots, 3$ se normalizan mediante:

$$R'(t_i) = \frac{R(t_i)}{\sqrt{\sum_n sw^2(n - t_i)}} \quad i = 1, \dots, 3 \quad (35)$$

El ganador de las tres correlaciones normalizadas se selecciona favoreciendo aquellos retardos que presenten valores en la gama inferior. Ello se logra ponderando las correlaciones normalizadas correspondientes a los retardos más largos. El mejor retardo de bucle abierto T_{op} se determina de la siguiente manera:

$$T_{op} = t_1$$

$$R'(T_{op}) = R'(t_1)$$

$$\text{si } R'(t_2) \geq 0,85R'(T_{op})$$

$$R'(T_{op}) = R'(t_2)$$

$$T_{op} = t_2$$

fin

$$\text{si } R'(t_3) \geq 0,85R'(T_{op})$$

$$R'(T_{op}) = R'(t_3)$$

$$T_{op} = t_3$$

fin

El recurso de dividir la gama de retardos en tres secciones para favorecer los valores menores tiene por objeto evitar la elección de los múltiplos del tono fundamental.

3.5 Cálculo de la respuesta de impulso

La respuesta a impulsos $h(n)$ del filtro de síntesis ponderado $W(z)/\hat{A}(z)$ se necesita para indagar las tablas de códigos adaptativos y fijos. Se calcula la respuesta de impulso $h(n)$ para cada subtrama filtrando una señal consistente en los coeficientes del filtro $A(z/\gamma_1)$ completado con ceros, a través de ambos filtros $1/\hat{A}(z)$ y $1/A(z/\gamma_2)$.

3.6 Cálculo de la señal objetivo

La señal objetivo $x(n)$ para la búsqueda de la tabla de códigos adaptativos suele calcularse restando la respuesta de entrada cero del filtro de síntesis ponderado $W(z)/\hat{A}(z) = A(z/\gamma_1)/[\hat{A}(z)A(z/\gamma_2)]$ de la señal vocal ponderada $sw(n)$ de la ecuación (33). Esto se realiza para cada subtrama.

Un procedimiento equivalente para calcular la señal objetivo, que se utiliza en esta Recomendación, consiste en filtrar la señal LP residual $r(n)$ a través de la combinación del filtro de síntesis $1/\hat{A}(z)$ y el filtro de ponderación $A(z/\gamma_1)/A(z/\gamma_2)$. Tras determinar la excitación correspondiente a la subtrama, los estados iniciales de dichos filtros se actualizan filtrando la diferencia entre las señales residual y de excitación. La actualización de la memoria de estos filtros se examina en 3.10.

La señal residual $r(n)$, necesaria para determinar el vector objetivo, también se aplica a la búsqueda de la tabla de códigos adaptativos, para ampliar la memoria intermedia de la excitación anterior. El recurso simplifica el procedimiento de búsqueda de la tabla de códigos adaptativos para retardos menores que el correspondiente a una subtrama de tamaño 40, como se verá en la próxima subcláusula. El residuo LP está definido por:

$$r(n) = s(n) + \sum_{i=1}^{10} \hat{a}_i s(n - i) \quad n = 0, \dots, 39 \quad (36)$$

3.7 Búsqueda de la tabla de códigos adaptativos

Los parámetros de tabla de códigos adaptativos (o parámetros de tono) son el retardo y la ganancia. Con arreglo al método de tabla de códigos adaptativos para aplicar el filtro de tono, se repite la excitación para retardos menores que la longitud de subtrama. En la fase de reconocimiento, la excitación se amplía mediante la LP residual, para simplificar la búsqueda en bucle cerrado. La búsqueda de la tabla de códigos adaptativos se efectúa para cada subtrama (5 ms). En la primera subtrama, se aplica un retardo de tono fraccionario T_1 con una definición de 1/3 en el intervalo $[19\frac{1}{3}, 84\frac{2}{3}]$ y únicamente con enteros en el intervalo [85, 143]. Para la segunda subtrama, se utiliza siempre un retardo T_2 con una definición de 1/3 en el intervalo $[int(T_1) - 5\frac{2}{3}, int(T_1) + 4\frac{2}{3}]$, en que $int(T_1)$ es la parte entera del retardo de tono fraccionario T_1 de la primera subtrama. Este intervalo se adapta para los casos en que T_1 sobrepasa los límites del intervalo de retardo.

Para cada subtrama, el retardo óptimo se determina mediante un análisis en bucle cerrado que minimiza el error cuadrático medio ponderado. En la primera subtrama, se determina el retardo T_1 investigando un pequeño intervalo (seis muestras) de valores de retardo en torno al retardo en bucle abierto T_{op} (véase 3.4). Los límites de la búsqueda $t_{mín}$ y $t_{máx}$ se definen mediante:

$$\begin{aligned} t_{mín} &= T_{op} - 3 \\ \text{si } t_{mín} < 20 \text{ entonces } t_{mín} &= 20 \\ t_{máx} &= t_{mín} + 6 \\ \text{si } t_{máx} > 143 \text{ entonces} \\ t_{máx} &= 143 \\ t_{mín} &= t_{máx} - 6 \\ \text{fin} \end{aligned}$$

Para la segunda subtrama, el análisis de tono en bucle cerrado se efectúa en torno al tono seleccionado de la primera subtrama, para determinar el retardo óptimo T_2 . El intervalo de búsqueda está limitado entre $t_{mín} - \frac{2}{3}$ y $t_{máx} + \frac{2}{3}$, donde $t_{mín}$ y $t_{máx}$ se deducen de T_1 como sigue:

$$\begin{aligned} t_{mín} &= int(T_1) - 5 \\ \text{si } t_{mín} < 20 \text{ entonces } t_{mín} &= 20 \\ t_{máx} &= t_{mín} + 9 \\ \text{si } t_{máx} > 143 \text{ entonces} \\ t_{máx} &= 143 \\ t_{mín} &= t_{máx} - 9 \\ \text{fin} \end{aligned}$$

La búsqueda de tono de bucle cerrado reduce al mínimo el error cuadrático medio ponderado entre la señal vocal original y la reconstruida. Esto se logra haciendo máximo el término:

$$R(k) = \frac{\sum_{n=0}^{39} x(n) y_k(n)}{\sqrt{\sum_{n=0}^{39} y_k(n) y_k(n)}} \quad (37)$$

donde $x(n)$ es la señal objetivo e $y_k(n)$ la excitación filtrada anterior en el retardo k [excitación anterior convolucionada con $h(n)$]. Obsérvese que el intervalo de búsqueda está limitado en torno a un valor preseleccionado, correspondiente al tono en bucle abierto T_{op} para la primera subtrama y T_1 para la segunda subtrama.

La convolución $y_k(n)$ se calcula para el retardo $t_{mín}$. Para los demás retardos enteros en el intervalo de búsqueda $k = t_{mín} + 1, \dots, t_{máx}$, se actualiza mediante la relación recursiva:

$$y_k(n) = y_{k-1}(n - 1) + u(-k)h(n) \quad n = 39, \dots, 0 \quad (38)$$

donde $u(n)$, $n = -143, \dots, 39$ es la memoria intermedia de excitación e $y_{k-1}(-1) = 0$. Obsérvese que, en la fase de búsqueda, las muestras $u(n)$, $n = 0, \dots, 39$ no se conocen y se necesitan para determinar los retardos de tono inferiores a 40. Para simplificar la búsqueda, se copia la LP residual a $u(n)$, de modo que la relación expresada en la ecuación (38) sea válida para todos los retardos.

Para determinar T_2 y T_1 cuando el retardo en bucle cerrado entero óptimo es inferior a 85, deben probarse las fracciones alrededor del retardo entero óptimo. La búsqueda de tono fraccionario se realiza interpolando la correlación normalizada de la ecuación (37) para encontrar su valor máximo. La interpolación tiene lugar mediante un filtro FIR b_{12} basado en una función sinc de Hamming ventanizada con truncamiento a ± 11 y completado con ceros a ± 12 [$b_{12}(12) = 0$]. El filtro tiene una frecuencia de corte (-3 dB) a 3600 Hz en el dominio de sobremuestra. Los valores interpolados de $R(k)$ para las fracciones $-\frac{2}{3}$, $-\frac{1}{3}$, 0 , $\frac{1}{3}$ y $\frac{2}{3}$ se obtienen aplicando la fórmula de interpolación:

$$R(k)_t = \sum_{i=0}^3 R(k - i)b_{12}(t + 3i) + \sum_{i=0}^3 R(k + 1 + i)b_{12}(3 - t + 3i) \quad t = 0, 1, 2 \quad (39)$$

en que $t = 0, 1, 2$ corresponde a las fracciones $0, \frac{1}{3}$ y $\frac{2}{3}$, respectivamente. Debe tenerse presente la necesidad de calcular los términos de correlación de la ecuación (37) utilizando el intervalo entre $t_{mín} - 4$ y $t_{máx} + 4$, para permitir una interpolación adecuada.

3.7.1 Generación del vector de tabla de códigos adaptativos

Una vez determinado el retardo de tono se calcula el vector de tabla de códigos adaptativos $v(n)$ interpolando la señal de excitación anterior $u(n)$ en el retardo entero dado k y la fracción t :

$$v(n) = \sum_{i=0}^9 u(n - k + i)b_{30}(t + 3i) + \sum_{i=0}^9 u(n - k + 1 + i)b_{30}(3 - t + 3i) \quad n = 0, \dots, 39 \quad t = 0, 1, 2 \quad (40)$$

El filtro de interpolación b_{30} se basa en una función sinc de Hamming ventanizada con truncamiento a ± 29 y se completa con ceros a ± 30 [$b_{30}(30) = 0$]. El filtro tiene una frecuencia de corte (-3 dB) a 3600 Hz en el dominio de sobremuestra.

3.7.2 Cálculo de palabras de código para retardos de tabla de códigos adaptativos

El retardo de tono T_1 se codifica con 8 bits en la primera subtrama y el retardo relativo en la segunda subtrama se codifica con 5 bits. El retardo fraccionario T está representado por su parte entera $int(T)$ y su parte fraccionaria $frac/3$, $frac = -1, 0, 1$. Se codifica entonces el índice de tono $P1$ según la relación:

$$P1 = \begin{cases} 3(int(T_1) - 19) + frac - 1 & \text{si } T_1 = [19, \dots, 85], frac = [-1, 0, 1] \\ (int(T_1) - 85) + 197 & \text{si } T_1 = [86, \dots, 143], frac = 0 \end{cases} \quad (41)$$

El valor del retardo de tono T_2 se codifica en relación con el valor de T_1 . Aplicando la misma interpretación arriba indicada, se codifica el retardo fraccionario T_2 , representado por su parte entera $int(T_2)$ y una parte fraccionaria $frac/3$, $frac = -1, 0, 1$, sobre la base de:

$$P2 = 3(int(T_2) - t_{\min}) + frac + 2 \quad (42)$$

donde t_{\min} se deriva de T_1 , igual que en 3.7.

Para que el códec sea más resistente a los errores de bit aleatorios, se calcula un bit de paridad $P0$ en el índice de retardo $P1$ de la primera subtrama. El bit de paridad se genera mediante una operación «o exclusivo» (XOR) en los seis bits más significativos de $P1$. En el decodificador este bit de paridad se vuelve a calcular, y si el valor nuevamente obtenido no coincide con el transmitido, se aplica un procedimiento de ocultamiento de errores.

3.7.3 Cálculo de la ganancia de tabla de códigos adaptativos

Después de determinar el retardo de tabla de códigos adaptativos, se calcula la ganancia de tabla de códigos adaptativos g_p según:

$$g_p = \frac{\sum_{n=0}^{39} x(n) y(n)}{\sum_{n=0}^{39} y(n) y(n)} \quad \text{en el intervalo } 0 \leq g_p \leq 1,2 \quad (43)$$

donde $x(n)$ es la señal objetivo e $y(n)$ es el vector de tabla de códigos adaptativos filtrado [respuesta al estado cero de $W(z)/\hat{A}(z)$ respecto de $v(n)$]. Este vector se obtiene mediante convolución de $v(n)$ con $h(n)$:

$$y(n) = \sum_{i=0}^n v(i)h(n - i) \quad n = 0, \dots, 39 \quad (44)$$

3.8 Tabla de códigos fijos: estructura y búsqueda

La tabla de códigos fijos se basa en una estructura de tabla de códigos algebraica mediante un diseño de permutación de impulso individual intercalado (ISPP, *interleaved single-pulse permutation*). En esta tabla, cada vector de tabla de códigos contiene cuatro impulsos no cero. Cada impulso puede tener amplitudes +1 ó -1 y asumir las posiciones que aparecen en el Cuadro 7.

Estructura de la tabla de códigos fijos \mathcal{C}

Impulso	Signo	Posiciones
i_0	$s_0: \pm 1$	$m_0: 0, 5, 10, 15, 20, 25, 30, 35$
i_1	$s_1: \pm 1$	$m_1: 1, 6, 11, 16, 21, 26, 31, 36$
i_2	$s_2: \pm 1$	$m_2: 2, 7, 12, 17, 22, 27, 32, 37$
i_3	$s_3: \pm 1$	$m_3: 3, 8, 13, 18, 23, 28, 33, 38$ 4, 9, 14, 19, 24, 29, 34, 39

El vector de tabla de códigos $c(n)$ se construye tomando un vector cero de dimensión 40 e introduciendo los cuatro impulsos unitarios en las posiciones encontradas, multiplicados por su signo correspondiente:

$$c(n) = s_0\delta(n - m_0) + s_1\delta(n - m_1) + s_2\delta(n - m_2) + s_3\delta(n - m_3) \quad n = 0, \dots, 39 \quad (45)$$

donde $\delta(0)$ es un impulso unitario. Una característica especial incorporada en la tabla de códigos es que el vector de tabla de códigos seleccionado se pasa a través de un prefiltro adaptativo $P(z)$ que amplía los componentes armónicos para mejorar la calidad de las señales vocales reconstruidas. Se utiliza para ello el filtro:

$$P(z) = 1 / (1 - \beta z^{-T}) \quad (46)$$

donde T es la componente entera del retardo de tono de la subtrama actual y β una ganancia de tono. El valor de β se hace adaptativo aplicando la cuantificación de la ganancia de tabla de códigos adaptativos de la subtrama anterior, es decir:

$$\beta = \hat{g}_p^{(m-1)} \quad \text{en el intervalo } 0,2 \leq \beta \leq 0,8 \quad (47)$$

Para retardos menores de 40, la tabla de códigos $c(n)$ de la ecuación (45) se modifica con arreglo a:

$$c(n) = \begin{cases} c(n) & n = 0, \dots, T - 1 \\ c(n) + \beta c(n - T) & n = T, \dots, 39 \end{cases} \quad (48)$$

Esta modificación se incorpora a la búsqueda de la tabla de códigos fijos modificando la respuesta a impulsos $h(n)$ según:

$$h(n) = \begin{cases} h(n) & n = 0, \dots, T - 1 \\ h(n) + \beta h(n - T) & n = T, \dots, 39 \end{cases} \quad (49)$$

3.8.1 Procedimiento de búsqueda de la tabla de códigos fijos

La tabla de códigos fijos se explora para hacer mínimo el error cuadrático medio entre la señal de voz ponderada de entrada $sw(n)$ de la ecuación (33) y la señal de voz ponderada reconstruida. Se actualiza la señal objetivo utilizada para la búsqueda del tono en bucle cerrado restando la contribución de la tabla de códigos adaptativos, o sea:

$$x'(n) = x(n) - g_p y(n) \quad n = 0, \dots, 39 \quad (50)$$

donde $y(n)$ es el vector de la tabla de códigos adaptativos de la ecuación (44) y g_p corresponde a la ganancia de la tabla de códigos adaptativos de la ecuación (43).

La matriz \mathbf{H} se define como la matriz inferior de convolución triangular de Toeplitz con diagonal $h(0)$ y diagonales inferiores $h(1), \dots, h(39)$. La matriz $\Phi = \mathbf{H}^t \mathbf{H}$ contiene las correlaciones de $h(n)$, mientras que los elementos de esta matriz simétrica están dados por:

$$\phi(i, j) = \sum_{n=j}^{39} h(n-i)h(n-j) \quad i = 0, \dots, 39 \quad j = i, \dots, 39 \quad (51)$$

La señal de correlación $d(n)$ se obtiene a partir de la señal objetivo $x'(n)$ y la respuesta a los impulsos $h(n)$ mediante:

$$d(n) = \sum_{i=n}^{39} x'(i)h(i-n) \quad n = 0, \dots, 39 \quad (52)$$

Si c_k representa el k -ésimo vector de la tabla de códigos fijos, la tabla de códigos se explora para hacer máxima la expresión:

$$\frac{C_k^2}{E_k} = \frac{\left(\sum_{n=0}^{39} d(n)c_k(n)\right)^2}{c_k^t \Phi c_k} \quad (53)$$

donde t denota transposición.

La señal $d(n)$ y la matriz Φ se calculan antes de buscar la tabla de códigos. Téngase presente que sólo se calculan los elementos estrictamente necesarios y que se ha diseñado un procedimiento eficaz de almacenamiento para acelerar la búsqueda.

La estructura algebraica de la tabla de códigos \mathcal{C} permite un procedimiento de búsqueda rápido, ya que el vector de códigos c_k contiene sólo cuatro impulsos no cero. La correlación del numerador de la ecuación (53) para un vector dado c_k viene dada por:

$$C = \sum_{i=0}^3 s_i d(m_i) \quad (54)$$

donde m_i es la posición del i -ésimo impulso y s_i es su amplitud. La energía del denominador de la ecuación (53) está dada por:

$$E = \sum_{i=0}^3 \phi(m_i, m_i) + 2 \sum_{i=0}^2 \sum_{j=i+1}^3 s_i s_j \phi(m_i, m_j) \quad (55)$$

Para simplificar el procedimiento de búsqueda, las amplitudes de impulso se determinan previamente cuantificando la señal $d(n)$. Para ello se establece que la amplitud de un impulso en determinada posición es igual al signo de $d(n)$ en dicha posición. Los siguientes pasos preceden a la búsqueda de la tabla de códigos. Se descompone primeramente la señal $d(n)$ en dos partes: su valor absoluto $|d(n)|$ y su signo $\text{sign}[d(n)]$. Seguidamente, se modifica la matriz Φ incorporando la información del signo, es decir:

$$\phi'(i, j) = \text{sign}[d(i)] \text{sign}[d(j)] \phi(i, j) \quad i = 0, \dots, 39 \quad j = i + 1, \dots, 39 \quad (56)$$

Los elementos de la diagonal principal de Φ se ponen a escala para eliminar el factor 2 de la ecuación (55):

$$\phi'(i, i) = 0,5\phi'(i, i) \quad i = 0, \dots, 39 \quad (57)$$

Ahora la correlación de la ecuación (54) tiene la forma:

$$C = |d(m_0)| + |d(m_1)| + |d(m_2)| + |d(m_3)| \quad (58)$$

y la energía de la ecuación (55) está dada por:

$$\begin{aligned} E/2 &= \phi'(m_0, m_0) \\ &+ \phi'(m_1, m_1) + \phi'(m_0, m_1) \\ &+ \phi'(m_2, m_2) + \phi'(m_0, m_2) + \phi'(m_1, m_2) \\ &+ \phi'(m_3, m_3) + \phi'(m_0, m_3) + \phi'(m_1, m_3) + \phi'(m_2, m_3) \end{aligned} \quad (59)$$

Para simplificar aún más el procedimiento de búsqueda, se aplica un método de búsqueda focalizada. Con arreglo a este método, se prueba un umbral previamente calculado antes de entrar el último bucle, que se introduce únicamente cuando dicho umbral se excede. El máximo número de veces en que el bucle puede incorporarse es fijo, de modo que se explora un porcentaje bajo de la tabla de códigos. El cálculo del umbral se basa en la correlación C . La correlación absoluta máxima y la correlación media, $máx_3$ y av_3 , resultantes de la contribución de los primeros tres impulsos, se determinan antes de indagar la tabla de códigos. El umbral se expresa por:

$$thr_3 = av_3 + K_3(máx_3 - av_3) \quad (60)$$

Se incorpora el cuarto bucle únicamente si la correlación absoluta (debida a tres impulsos) excede thr_3 , donde $0 \leq K_3 < 1$. El valor de K_3 , determina el porcentaje de búsqueda de la tabla de códigos y se fija en este caso en 0,4. Obsérvese que esto da lugar a un tiempo de búsqueda variable. Para verificar posteriormente la búsqueda, el número de veces que se incorpora el último bucle (para ambas subtramas) no puede sobrepasar un cierto máximo, que aquí se establece en 180 (el promedio para el caso peor por subtrama es 90 veces).

3.8.2 Cálculo de palabra de código de la tabla de códigos fijos

Las posiciones de los impulsos i_0 , i_1 , e i_2 se codifican con 3 bits cada una, en tanto que la posición de i_3 se codifica con 4 bits. La amplitud de cada impulso se codifica con 1 bit. Esto proporciona un total de 17 bits para los cuatro impulsos. Definiendo $s = 1$ cuando el signo es positivo y $s = 0$ cuando el signo es negativo, la palabra de código del signo se obtiene de:

$$S = s_0 + 2s_1 + 4s_2 + 8s_3 \quad (61)$$

y la palabra de código de la tabla de códigos fijos se obtiene de:

$$C = (m_0/5) + 8(m_1/5) + 64(m_2/5) + 512(2(m_3/5) + jx) \quad (62)$$

donde $jx = 0$ cuando $m_3 = 3, 8, \dots, 38$, y $jx = 1$ cuando $m_3 = 4, 9, \dots, 39$.

3.9 Cuantificación de las ganancias

La ganancia de las tablas de códigos adaptativos (ganancia de tono) y la ganancia de la tabla de códigos fijos son vectores cuantificados con 7 bits. La búsqueda de la tabla de códigos de ganancia se lleva a cabo haciendo mínimo el error cuadrático medio ponderado entre la señal de voz original y la señal reconstruida, dado por:

$$E = \mathbf{x}^t \mathbf{x} + g_p^2 \mathbf{y}^t \mathbf{y} + g_c^2 \mathbf{z}^t \mathbf{z} - 2g_p \mathbf{x}^t \mathbf{y} - 2g_c \mathbf{x}^t \mathbf{z} + 2g_p g_c \mathbf{y}^t \mathbf{z} \quad (63)$$

donde \mathbf{x} es el vector objetivo (véase 3.6), \mathbf{y} es el vector de la tabla de códigos adaptativos filtrado de la ecuación (44) y \mathbf{z} es el vector de la tabla de códigos fijos convolucionado con $h(n)$,

$$z(n) = \sum_{i=0}^n c(i)h(n-i) \quad n = 0, \dots, 39 \quad (64)$$

3.9.1 Predicción de la ganancia

La ganancia de la tabla de códigos fijos g_c puede expresarse como

$$g_c = \gamma g'_c \quad (65)$$

donde g'_c es una ganancia predicha en base a las energías anteriores de tabla de códigos fijos y γ es un factor de corrección.

La energía media de la contribución de la tabla de códigos fijos está dada por:

$$E = 10 \log \left(\frac{1}{40} \sum_{n=0}^{39} c(n)^2 \right) \quad (66)$$

Después de poner a escala el vector $c(n)$ con la ganancia de tabla de códigos fijos g_c , la energía de tabla de códigos fijos en escala viene dada por $20 \log g_c + E$. Sea $E^{(m)}$ la energía eliminada media (en dB) de la contribución (a escala) de tabla de códigos fijos en la subtrama m , dada por:

$$E^{(m)} = 20 \log g_c + E - \bar{E} \quad (67)$$

donde $\bar{E} = 30$ dB es la energía media de la excitación de la tabla de códigos fijos. La ganancia g_c puede expresarse como una función de $E^{(m)}$, E y \bar{E} mediante:

$$g_c = 10^{(E^{(m)} + \bar{E} - E)/20} \quad (68)$$

La ganancia dicha g'_c se determina prediciendo la energía logarítmica de la contribución de la tabla de código fijo actual a partir de la energía logarítmica de las contribuciones de tabla de códigos fijos anteriores. La predicción de MA de 4.º orden se efectúa como sigue. La energía predicha está dada por:

$$\tilde{E}^{(m)} = \sum_{i=1}^4 b_i \hat{U}^{(m-i)} \quad (69)$$

donde $[b_1 \ b_2 \ b_3 \ b_4] = [0,68 \ 0,58 \ 0,34 \ 0,19]$ son los coeficientes de predicción de MA y $\hat{U}^{(m)}$ es la versión cuantificada del error de predicción $U^{(m)}$ en la subtrama m , definido por:

$$U^{(m)} = E^{(m)} - \tilde{E}^{(m)} \quad (70)$$

La ganancia predicha g'_c se determina sustituyendo $E^{(m)}$ por su valor predicho en la ecuación (68):

$$g'_c = 10^{(\tilde{E}^{(m)} + \bar{E} - E)/20} \quad (71)$$

El factor de corrección γ se relaciona con el error de predicción de ganancia mediante:

$$U^{(m)} = E^{(m)} - \tilde{E}^{(m)} = 20 \log(\gamma) \quad (72)$$

3.9.2 Búsqueda de la tabla de códigos para cuantificación de la ganancia

La ganancia de tabla de códigos adaptativos g_p y el factor γ se cuantifican vectorialmente mediante una tabla de códigos con estructura conjugada de dos fases. La primera fase consiste en una tabla de códigos bidimensional de 3 bits $\mathcal{Y}\mathcal{A}$, y la segunda en una tabla de códigos bidimensional de 4 bits $\mathcal{Y}\mathcal{B}$. El primer elemento en cada tabla de códigos representa la ganancia cuantificada de tabla de códigos adaptativos \hat{g}_p , y el segundo elemento el factor $\hat{\gamma}$ de corrección de ganancia cuantificada de tabla de códigos fijos. Dados los índices de tabla de códigos GA y GB para $\mathcal{Y}\mathcal{A}$ y $\mathcal{Y}\mathcal{B}$ respectivamente, la ganancia cuantificada de tabla de códigos adaptativos se determina mediante:

$$\hat{g}_p = \mathcal{Y}\mathcal{A}_1(GA) + \mathcal{Y}\mathcal{B}_1(GB) \quad (73)$$

mientras que la ganancia cuantificada de la tabla de códigos fijos viene dada por:

$$\hat{g}_c = g'_c \hat{\gamma} = g'_c(\mathcal{Y}\mathcal{A}_2(GA) + \mathcal{Y}\mathcal{B}_2(GB)) \quad (74)$$

Esta estructura conjugada simplifica la indagación de la tabla de códigos al introducir un proceso de preselección. La ganancia óptima de tono g_p y la ganancia de tabla de códigos fijos g_c se obtienen a partir de la ecuación (63) y se utilizan para la preselección. La tabla de códigos $\mathcal{Y}\mathcal{A}$ contiene ocho entradas en las cuales el segundo elemento (correspondiente a g_c) tiene generalmente valores mayores que el primer elemento (correspondiente a g_p). Esta desviación permite una preselección utilizando el valor del g_c . En este proceso de preselección, se elige un grupo de cuatro vectores cuyos segundos elementos son cercanos a g_c . De manera similar, la tabla de códigos $\mathcal{Y}\mathcal{B}$ contiene 16 entradas con una desviación hacia el primer elemento (correspondiente a g_p). Se elige un grupo de ocho vectores cuyos primeros elementos son cercanos a g_p . De este modo, se selecciona para cada tabla de códigos el mejor 50% de vectores candidatos. Seguidamente, se lleva a cabo una investigación exhaustiva de las restantes 32 posibilidades (4×8), de modo que la combinación de ambos índices minimice el error cuadrático medio ponderado de la ecuación (63).

3.9.3 Cálculo de palabras de código para el cuantificador de ganancia

Las palabras de código GA y GB para el cuantificador de ganancia se obtienen a partir de los índices correspondientes a la mejor selección. Con el fin de reducir las repercusiones de los errores de bit individuales, se comparan los índices de tablas de códigos.

3.10 Actualización de la memoria

Es necesario actualizar los estados de los filtros de síntesis y ponderación para calcular la señal objetivo en la subtrama siguiente. Después de cuantificar las dos ganancias, la señal de excitación, $u(n)$, en la subtrama actual se obtiene mediante:

$$u(n) = \hat{g}_p v(n) + \hat{g}_c c(n) \quad n = 0, \dots, 39 \quad (75)$$

donde \hat{g}_p y \hat{g}_c son las ganancias cuantificadas de las tablas de códigos adaptativos y fijos respectivamente, $v(n)$ es el vector de tabla de códigos adaptativos (excitación anterior interpolada) y $c(n)$ es el vector de tabla de códigos fijos que incluye los armónicos ampliados. Los estados de los filtros pueden actualizarse filtrando la señal $r(n)-u(n)$ (diferencia entre residuo y excitación) a través de los filtros $1/\hat{A}(z)$ y $A(z/\gamma_1)/A(z/\gamma_2)$ para la subtrama de 40 muestras y conservando los estados de los filtros. Ello requiere tres operaciones de filtrado. Un método más simple, con una sola operación de filtrado, se describe a continuación. La señal vocal reconstruida localmente $\hat{s}(n)$ se calcula filtrando la señal de excitación a través de $1/\hat{A}(z)$. La salida del filtro correspondiente a la entrada $r(n) - u(n)$ equivale a $e(n) = s(n) - \hat{s}(n)$. De tal suerte, los estados del filtro de síntesis $1/\hat{A}(z)$ están determinados por $e(n)$, $n = 30, \dots, 39$. La actualización de los estados del filtro $A(z/\gamma_1)/A(z/\gamma_2)$ puede efectuarse filtrando la señal de error $e(n)$ a través de dicho filtro, para determinar el error ponderado perceptualmente $ew(n)$. No obstante, la señal $ew(n)$ puede igualmente encontrarse mediante:

$$ew(n) = x(n) - \hat{g}_p y(n) - \hat{g}_c z(n) \quad (76)$$

Disponiendo de las señales $x(n)$, $y(n)$ y $z(n)$, los estados del filtro de ponderación se actualizan calculando $ew(n)$ como se indica en la ecuación (76), para $n = 30, \dots, 39$. Este procedimiento ahorra dos operaciones de filtrado.

4 Descripción de las funciones del decodificador

El principio del decodificador se describió en la cláusula 2 (Figura 3). Primero se decodifican los parámetros (coeficientes LP, vector de tabla de códigos adaptativos, vector de tabla de códigos fijos y ganancias). En el Cuadro 8 se enumeran los parámetros transmitidos. Estos parámetros decodificados se utilizan para calcular la señal vocal reconstruida, tal como se verá en 4.1. La señal reconstruida se mejora mediante una operación de postprocesamiento consistente en un postfiltro, un filtro de paso alto y un escalamiento ascendente (véase 4.2). En 4.4 se describe el procedimiento de ocultamiento de errores, aplicable cuando tiene lugar un error de paridad o bien se ha establecido una bandera de borrado de trama. La Figura 6 presenta un diagrama detallado del flujo de señales en el decodificador.

CUADRO 8/G.729

Descripción de los índices de parámetros transmitidos – El ordenamiento del tren de bits corresponde al orden establecido en el cuadro – Para cada parámetro, el bit más significativo (MSB, *most significant bit*) es el que se transmite primero

Símbolo	Descripción	Bits
<i>L0</i>	Índice del predictor MA conmutado del cuantificador LSP	1
<i>L1</i>	Vector de la primera fase del cuantificador LSP	7
<i>L2</i>	Vector de la segunda fase inferior del cuantificador LSP	5
<i>L3</i>	Vector de la segunda fase superior del cuantificador LSP	5
<i>P1</i>	Primera subtrama del retardo de tono	8
<i>P0</i>	Bit de paridad para el retardo de tono	1
<i>C1</i>	Primera subtrama de la tabla de códigos fijos	13
<i>S1</i>	Signos de la primera subtrama de los impulsos de la tabla de códigos fijos	4
<i>GA1</i>	Primera subtrama de la tabla de códigos de ganancia (fase 1)	3
<i>GB1</i>	Primera subtrama de la tabla de códigos de ganancia (fase 2)	4
<i>P2</i>	Segunda subtrama del retardo de tono	5
<i>C2</i>	Segunda subtrama de la tabla de códigos fijos	13
<i>S2</i>	Signos de la segunda subtrama de los impulsos de la tabla de códigos fijos	4
<i>GA2</i>	Segunda subtrama de la tabla de códigos de ganancia (fase 1)	3
<i>GB2</i>	Segunda subtrama de la tabla de códigos de ganancia (fase 2)	4

4.1 Procedimiento de decodificación de los parámetros

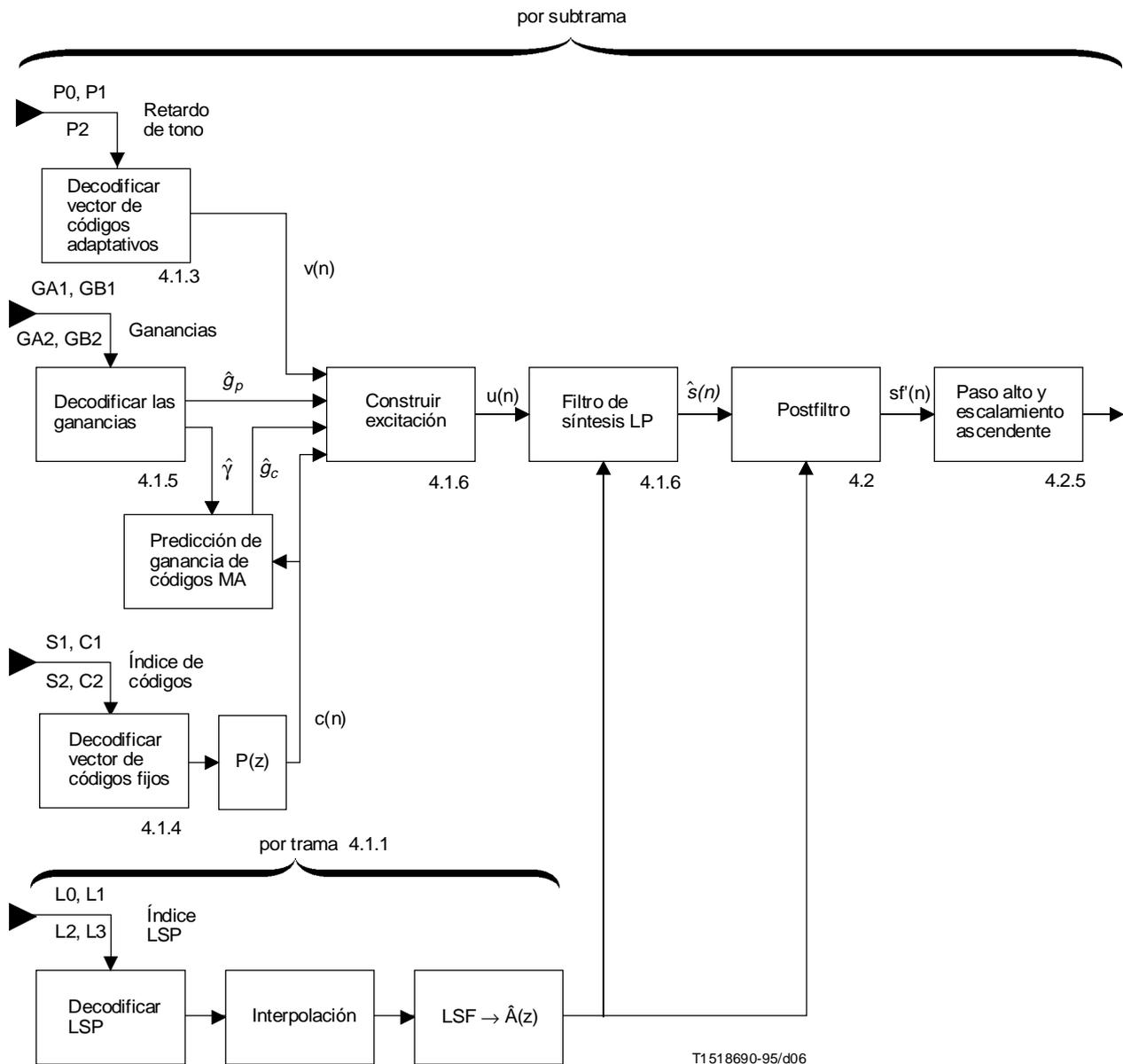
El proceso de decodificación tiene lugar en el siguiente orden.

4.1.1 Decodificación de los parámetros de filtro LP

Los índices recibidos *L0*, *L1*, *L2* y *L3* del cuantificador LSP se utilizan para reconstruir los coeficientes LSP cuantificados, mediante el procedimiento descrito en 3.2.4. Se aplica el procedimiento de interpolación descrito en 3.2.5 para obtener dos conjuntos de coeficientes LSP interpolados (correspondientes a dos subtramas). Para cada subtrama, los coeficientes LSP interpolados se convierten en coeficientes a_i del filtro LP, que se utilizan para sintetizar la señal vocal reconstruida en la subtrama.

Se repiten los siguientes pasos para cada subtrama:

- 1) decodificar el vector de tabla de códigos adaptativos;
- 2) decodificar el vector de tabla de códigos fijos;
- 3) decodificar las ganancias de las tablas de códigos adaptativos y fijos;
- 4) calcular la señal vocal reconstruida.



T1518690-95/d06

FIGURA 6/G.729
Flujo de señales en el decodificador CS-ACELP

4.1.2 Cálculo del bit de paridad

Antes de reconstruir la excitación, se calcula nuevamente el bit de paridad a partir del índice de retardo $P1$ de la tabla de códigos adaptativos (véase 3.7.2). Si dicho bit no coincide con el bit de paridad transmitido $P0$, es probable que se hayan producido errores de transmisión.

En caso de producirse un error de paridad en $P1$, se fija el valor de retardo T_1 como la parte entera del valor de retardo T_2 de la trama precedente. El valor de T_2 se deriva del procedimiento desarrollado en 4.1.3, introduciendo el nuevo valor de T_1 .

4.1.3 Decodificación del vector de tabla de códigos adaptativos

Si no ha habido error de paridad, el índice $P1$ de tabla de códigos adaptativos se utiliza para determinar las partes entera y fraccionaria del retardo de tono T_1 . La parte entera $int(T_1)$ y la parte fraccionaria de T_1 frac se obtienen a partir de $P1$ como se indica a continuación:

si $P1 < 197$

$$int(T_1) = (P1 + 2)/3 + 19$$

$$frac = P1 - 3 int(T_1) + 58$$

de otro modo

$$int(T_1) = P1 - 112$$

$$frac = 0$$

fin

Las partes entera y fraccionaria de T_2 se obtienen a partir de $P2$ y $t_{mín}$, donde $t_{mín}$ se deriva de T_1 como sigue:

$$t_{mín} = int(T_1) - 5$$

$$si t_{mín} < 20 entonces t_{mín} = 20$$

$$t_{máx} = t_{mín} + 9$$

$$si t_{máx} > 143 entonces$$

$$t_{máx} = 143$$

$$t_{mín} = t_{máx} - 9$$

fin

Ahora se decodifica T_2 mediante:

$$int(T_2) = (P2 + 2)/3 - 1 + t_{mín}$$

$$frac = P2 - 2 - 3 ((P2 + 2)/3 - 1)$$

El vector de tabla de códigos adaptativos $v(n)$ se encuentra interpolando la excitación anterior $u(n)$ (en el retardo de tono) mediante la ecuación (40).

4.1.4 Decodificación del vector de tabla de códigos fijos

El índice recibido C de tabla de códigos fijos se utiliza para extraer las posiciones de los impulsos de excitación. Los signos de los impulsos se obtienen a partir de S . Para ello se invierte el proceso descrito en 3.8.2. Una vez decodificados los signos y las posiciones de los impulsos, se construye el vector $c(n)$ de tabla de códigos fijos de acuerdo con la ecuación (45). Si la parte entera del retardo de tono T es inferior al tamaño de la subtrama, fijado en 40, se modifica $c(n)$ con arreglo a la ecuación (48).

4.1.5 Decodificación de las ganancias de las tablas de códigos adaptativos y fijos

El índice recibido de tabla de códigos de ganancia determina la ganancia \hat{g}_p de tabla de códigos adaptativos, así como el factor de corrección \hat{g} de la ganancia de tabla de códigos fijos. El procedimiento se describe en 3.9. Se calcula la ganancia estimada g'_c de la tabla de códigos fijos mediante la ecuación (71). El vector de tabla de códigos fijos se obtiene a partir del producto del factor de corrección de ganancia cuantificado y por la ganancia predicha, según la ecuación (74). La ganancia de tabla de códigos adaptativos se reconstruye mediante la ecuación (73).

4.1.6 Cálculo de la señal vocal reconstruida

La excitación $u(n)$ [véase la ecuación (75)] entra al filtro de síntesis LP. La señal de voz reconstruida para la subtrama está dada por:

$$\hat{s}(n) = u(n) - \sum_{i=1}^{10} \hat{a}_i \hat{s}(n - i) \quad n = 0, \dots, 39 \quad (77)$$

donde \hat{a}_i son los coeficientes interpolados del filtro LP para la subtrama actual. La señal vocal reconstruida $\hat{s}(n)$ es luego procesada mediante el postprocesador que se describe en la subcláusula siguiente.

4.2 Postprocesamiento

El postprocesamiento consta de tres funciones: postfiltrado adaptativo, filtrado de paso alto y aplicación de un escalamiento ascendente a las señales. El postfiltro adaptativo es una cascada de tres filtros: un postfiltro de largo plazo $H_p(z)$, un postfiltro de corto plazo $H_f(z)$ y como un filtro de compensación de pendiente $H_t(z)$, seguidos de un procedimiento de control de ganancia adaptativo. Los coeficientes del postfiltro se actualizan cada subtrama de 5 ms. El proceso de postfiltrado se organiza de la siguiente manera. En primer lugar, la voz reconstruida $\hat{s}(n)$ es filtrada inversamente a través de $\hat{A}(z/\gamma_n)$, produciendo la señal residual $\hat{r}(n)$. Esta señal se utiliza para calcular el retardo T y la ganancia g_t del filtro de largo plazo $H_p(z)$. La señal $\hat{r}(n)$ se filtra entonces a través del postfiltro de largo plazo $H_p(z)$ y el filtro de síntesis $1/[g_f \hat{A}(z/\gamma_d)]$. Por último, la señal de salida del filtro de síntesis $1/[g_f \hat{A}(z/\gamma_d)]$ se pasa a través del filtro de compensación de pendiente $H_t(z)$, para generar la señal vocal reconstruida postfiltrada $sf(n)$. El control de ganancia adaptativo se aplica entonces a $sf(n)$ para ajustar la energía de $\hat{s}(n)$. La señal resultante $sf'(n)$ pasa por un filtro de paso alto y se escalona para producir la señal de salida del decodificador.

4.2.1 Postfiltro de largo plazo

El postfiltro de largo plazo está dado por:

$$H_p(z) = \frac{1}{1 + \gamma_p g_l z^{-T}} \quad (78)$$

donde T es el retardo de tono y g_l el coeficiente de ganancia. Obsérvese que g_l es como máximo 1 y que su valor se fija en cero si la ganancia de predicción a largo plazo es inferior a 3 dB. El factor γ_p controla la cantidad de postfiltrado a largo plazo y tiene el valor $\gamma_p = 0,5$. El retardo y la ganancia a largo plazo se calculan a partir de la señal residual $\hat{r}(n)$, que se obtiene al filtrar la señal vocal $\hat{s}(n)$ a través de $\hat{A}(z/\gamma_n)$, numerador del postfiltro de corto plazo (véase 4.2.2).

$$\hat{r}(n) = \hat{s}(n) + \sum_{i=1}^{10} \gamma_n^i \hat{a}_i \hat{s}(n - i) \quad (79)$$

El retardo de largo plazo se calcula en dos pasos. El primer paso selecciona el mejor entero T_0 en el intervalo $[int(T_1) - 1, int(T_1) + 1]$, siendo $int(T_1)$ la parte entera del retardo (transmitido) de tono T_1 en la primera subtrama. El retardo entero mejor es el que hace máxima la correlación:

$$R(k) = \sum_{n=0}^{39} \hat{r}(n) \hat{r}(n - k) \quad (80)$$

El segundo paso selecciona el mejor retardo fraccionario T con una definición de 1/8 en torno a T_0 . Esto se obtiene buscando el retardo con la mayor correlación pseudonormalizada:

$$R'(k) = \frac{\sum_{n=0}^{39} \hat{r}(n) \hat{r}_k(n)}{\sqrt{\sum_{n=0}^{39} \hat{r}_k(n) \hat{r}_k(n)}} \quad (81)$$

donde $\hat{r}_k(n)$ es la señal residual en el retardo k . Obtenido el retardo óptimo T , la correlación correspondiente $R'(T)$ se normaliza respecto de la raíz cuadrada de la energía de $\hat{r}(n)$. El cuadrado de esta correlación normalizada se utiliza para determinar si el postfiltro de largo plazo debe desconectarse. Para ello se introduce $g_l = 0$ cuando:

$$\frac{R'(T)^2}{\sum_{n=0}^{39} \hat{r}(n)\hat{r}(n)} < 0,5 \quad (82)$$

De otro modo, el valor de g_l se calcula a partir de:

$$g_l = \frac{\sum_{n=0}^{39} \hat{r}(n)\hat{r}_k(n)}{\sum_{n=0}^{39} \hat{r}_k(n)\hat{r}_k(n)} \quad \text{en el intervalo } 0 \leq g_l \leq 1,0 \quad (83)$$

La señal retardada no entera $\hat{r}_k(n)$ es el primer valor calculado mediante un filtro de interpolación de longitud 33. Después de seleccionar T , se calcula de nuevo $\hat{r}_k(n)$ mediante un filtro de interpolación más largo, de longitud 129. La nueva señal reemplaza a la anterior solamente en el caso en que el filtro más largo aumente el valor de $R'(T)$.

4.2.2 Postfiltro de corto plazo

El postfiltro de corto plazo está dado por:

$$H_f(z) = \frac{1}{g_f} \frac{\hat{A}(z/\gamma_n)}{\hat{A}(z/\gamma_d)} = \frac{1}{g_f} \frac{1 + \sum_{i=1}^{10} \gamma_n^i \hat{a}_i z^{-i}}{1 + \sum_{i=1}^{10} \gamma_d^i \hat{a}_i z^{-i}} \quad (84)$$

donde $\hat{A}(z)$ es el filtro LP inverso cuantificado recibido (no hay análisis LP en el decodificador), mientras que los factores γ_n y γ_d controlan la cantidad de postfiltrado de corto plazo, fijándose en $\gamma_n = 0,55$ y $\gamma_d = 0,7$. El término de ganancia g_f se calcula en la respuesta a los impulsos truncada $h_f(n)$ del filtro $\hat{A}(z/\gamma_n)/\hat{A}(z/\gamma_d)$ y viene dado por:

$$g_f = \sum_{n=0}^{19} |h_f(n)| \quad (85)$$

4.2.3 Compensación de la pendiente

El filtro $H_t(z)$ compensa la pendiente en el postfiltro de corto plazo $H_f(z)$ y viene dado por:

$$H_t(z) = \frac{1}{g_t} (1 + \gamma_t k'_1 z^{-1}) \quad (86)$$

donde $\gamma_t k'_1$ es el factor de pendiente, siendo k'_1 el primer coeficiente de reflexión calculado a partir de $h_f(n)$ para:

$$k'_1 = -\frac{r_h(1)}{r_h(0)} \quad r_h(i) = \sum_{j=0}^{19-i} h_f(j) h_f(j+i) \quad (87)$$

El término de ganancia $g_t = 1 - |\gamma_t k'_1|$ compensa el efecto decreciente de g_f en $H_f(z)$. Se ha visto además que el filtro producto $H_f(z)H_t(z)$ no suele generar ganancia. Dos valores de γ_t se aplican según el signo de k'_1 . Cuando k'_1 es negativo, $\gamma_t = 0,9$, cuando k'_1 es positivo, $\gamma_t = 0,2$.

4.2.4 Control de ganancia adaptativo

Se aplica el control de ganancia adaptativo para compensar las diferencias de ganancia entre la señal vocal reconstruida $\hat{s}(n)$ y la señal postfiltrada $sf(n)$. El factor de escala de ganancia G para la subtrama actual se calcula mediante:

$$G = \frac{\sum_{n=0}^{39} |\hat{s}(n)|}{\sum_{n=0}^{39} |sf(n)|} \quad (88)$$

La señal de ganancia postfiltrada a escala $sf'(n)$ está dada por:

$$sf'(n) = g^{(n)}sf(n) \quad n = 0, \dots, 39 \quad (89)$$

donde $g^{(n)}$ se actualiza para cada muestra y su valor está determinado por:

$$g^{(n)} = 0,85 g^{(n-1)} + 0,15 G \quad n = 0, \dots, 39 \quad (90)$$

Se utiliza el valor inicial de $g^{(-1)} = 1,0$. Seguidamente, para cada nueva subtrama, $g^{(-1)}$ se iguala a $g^{(39)}$ de la subtrama anterior.

4.2.5 Filtrado de paso alto y escalamiento ascendente

Se aplica un filtro de paso alto con una frecuencia de corte de 100 Hz a la señal de voz postfiltrada reconstruida $sf'(n)$. El filtro viene dado por:

$$H_{h2}(z) = \frac{0,93980581 - 1,8795834z^{-1} + 0,93980581z^{-2}}{1 - 1,9330735z^{-1} + 0,93589199z^{-2}} \quad (91)$$

La señal filtrada se multiplica por 2 para restaurar el nivel de la señal de entrada.

4.3 Inicialización del codificador y el decodificador

Con excepción de las variables del Cuadro 9 que sigue, todas las demás variables estáticas del codificador y del decodificador deben fijarse inicialmente a 0.

CUADRO 9/G.729

Descripción de parámetros con valores de inicialización distintos de 0

Variable	Referencia	Valor inicial
β	3.8	0,8
$g^{(-1)}$	4.2.4	1,0
$\hat{\lambda}_i$	3.2.4	$i\pi/11$
q_i	3.2.4	$\arccos(i\pi/11)$
$\hat{U}^{(k)}$	3.9.1	-14

4.4 Ocultamiento de borrados de tramas

Se ha incorporado al decodificador un procedimiento de ocultamiento de errores para reducir la degradación de la señal vocal reconstruida debida a los borrados de trama en el tren de bits. Este proceso de ocultamiento de errores se aplica toda vez que se haya detectado el borrado de una trama de parámetros del códec (correspondientes a una trama de 10 ms). La Recomendación no define el mecanismo para detectar los borrados de tramas, mecanismo que dependerá de la aplicación.

La estrategia de ocultamiento consiste en reconstruir la trama actual en base a la información recibida anteriormente. Según este método, se reemplaza la señal de excitación que falta por otra de características similares pero con energía levemente inferior. Se utiliza para ello un clasificador vocal basado en la ganancia de predicción a largo plazo, calculada en el marco del análisis del postfiltro de largo plazo. El postfiltro de largo plazo (véase 4.2.1) halla el predictor de largo plazo para el cual la ganancia de predicción sea mayor que 3 dB. Para ello se establece un umbral de 0,5 en la correlación cuadrática normalizada de la ecuación (82). A los efectos de procesar el ocultamiento de errores, una trama de 10 ms se declara periódica si al menos una subtrama de 5 ms tiene una ganancia de predicción a largo plazo de más de 3 dB. En los demás casos la trama se declara no periódica. Toda trama borrada conservará la clase de la trama vocal (reconstruida) precedente. Obsérvese que la clasificación vocal es actualizada en forma permanente sobre la base de la señal vocal reconstruida.

Los pasos concretos que se dan para una trama borrada son:

- 1) repetición de los parámetros del filtro de síntesis;
- 2) atenuación de las ganancias de tablas de códigos adaptativos y fijos;
- 3) atenuación de la memoria del predictor de ganancia;
- 4) generación de la excitación de reemplazo.

4.4.1 Repetición de los parámetros del filtro de síntesis

El filtro de síntesis en el caso de una trama borrada utiliza los parámetros LP de la última trama buena. La memoria del predictor LSF de MA contiene los valores de las palabras de código recibidas \hat{l}_i . Puesto que no se dispone de la palabra de código para la trama actual m , ésta se calcula a partir de los parámetros repetidos $\hat{\omega}_i$ de LSF y de la memoria del predictor, mediante:

$$\hat{l}_i = \left[\hat{\omega}_i^{(m)} - \sum_{k=1}^4 \hat{p}_{i,k} \hat{l}_i^{(m-k)} \right] / \left(1 - \sum_{k=1}^4 \hat{p}_{i,k} \right) \quad i = 1, \dots, 10 \quad (92)$$

donde los coeficientes $\hat{p}_{i,k}$ del predictor MA son los de la última trama buena recibida.

4.4.2 Atenuación de las ganancias de las tablas de códigos adaptativos y fijos

La ganancia de tabla de códigos fijos se basa en una versión atenuada de la ganancia de tabla de códigos fijos anterior, de acuerdo con:

$$g_c^{(m)} = 0,98g_c^{(m-1)} \quad (93)$$

donde m es el índice de subtrama. La ganancia de tabla de códigos adaptativos se basa en una versión atenuada de la ganancia de tabla de códigos adaptativos anterior, de acuerdo con:

$$g_p^{(m)} = 0,9g_p^{(m-1)} \quad \text{con la condición de que } g_p^{(m)} < 0,9 \quad (94)$$

4.4.3 Atenuación de la memoria del predictor de ganancias

De conformidad con lo descrito en 3.9, el predictor de ganancia utiliza la energía de los vectores $c(n)$ previamente seleccionados de tabla de códigos fijos. Para evitar los efectos de transición en el decodificador, cuando se reciben tramas intactas, la memoria del predictor de ganancias se actualiza mediante una versión atenuada de la energía de la tabla de códigos. El valor $\hat{U}^{(m)}$ para la subtrama actual m se pone al valor del error de predicción de ganancia cuantificada media, atenuado en 4 dB:

$$\hat{U}^{(m)} = \left(0,25 \sum_{i=1}^4 \hat{U}^{(m-i)} \right) - 4,0 \quad \text{con la condición de que } \hat{U}^{(m)} \geq -14 \quad (95)$$

4.4.4 Generación de la excitación de reemplazo

La excitación utilizada depende de la clasificación de periodicidad. Si la última trama reconstruida se calificó de periódica, también se considerará periódica la trama actual. En tal caso sólo se aplicará la tabla de códigos adaptativos, poniéndose a cero la contribución de la tabla de códigos fijos. El retardo de tono se basa en la parte entera de dicho retardo de la trama anterior, y se repite para cada trama sucesiva. Para evitar una repetición excesiva, se aumenta el retardo a razón de una unidad por cada subtrama siguiente, hasta un máximo de 143. La ganancia de tabla de códigos adaptativos se basa en un valor atenuado con arreglo a la ecuación (94).

Si la última trama reconstruida se calificó de no periódica, también se considerará no periódica la trama actual, estableciéndose en cero la contribución de la tabla de códigos adaptativos. La contribución de la tabla de códigos fijos se establece mediante la selección aleatoria de un índice de la tabla de códigos y un índice de signo. El generador aleatorio se basa en la función:

$$seed = 31821 seed + 13849 \quad (96)$$

con un valor seed inicial de 21845. El índice de tabla de códigos fijos se deriva de los 13 bits menos significativos del número aleatorio siguiente. El signo de tabla de códigos fijos se deriva de los 4 bits menos significativos del número aleatorio siguiente. La ganancia de tabla de códigos fijos se atenúa con arreglo a la ecuación (93).

5 Descripción binaria exacta del códec CS-ACELP

En el UIT-T puede obtenerse el código C de ANSI para simular un códec CS-ACELP de coma fija de 16 bits. En las subcláusulas siguientes se presenta un resumen del empleo de este código de simulación, así como la organización del correspondiente soporte lógico.

5.1 Empleo del soporte lógico de simulación

El código C consta de dos programas principales: **coder.c**, que simula el codificador, y **decoder.c** que simula el decodificador. El codificador se pone en marcha como sigue:

coder inputfile bitstreamfile

Los ficheros de entrada y de salida son ficheros de datos de muestreo que contienen señales MIC a 16 bits. El decodificador se pone en marcha como sigue:

decoder bitstreamfile outputfile

El cuadro de correspondencia del tren codificado está contenido en el soporte lógico de simulación.

5.2 Organización del soporte lógico de simulación

En la simulación C de ANSI de coma fija sólo se utilizan dos tipos de datos de coma fija, como se ve en el Cuadro 10. Con el fin de facilitar la aplicación del código de simulación, índices de bucle, valores booleanos y banderas, se emplea el tipo **Flag**, que será de 16 ó 32 bits, según sea la plataforma objetivo.

CUADRO 10/G.729

Tipos de datos utilizados en la simulación C de ANSI

Tipo	Valor máximo	Valor mínimo	Descripción
Word16	0x7fff	0x8000	Palabra de 16 bits con complemento de 2 con signo
Word32	0x7fffffffL	0x80000000L	Palabra de 32 bits con complemento de 2 con signo

Todos los cálculos se efectúan empleando un conjunto predefinido de operaciones básicas. La descripción de dichas operaciones aparece en el Cuadro 11. Las tablas utilizadas por el códec de simulación se resumen en el Cuadro 12. Estos programas principales emplean una biblioteca de subprogramas que se resumen en los Cuadros 13, 14 y 15.

CUADRO 11/G.729

Operaciones básicas empleadas en la simulación C de ANSI

Operación	Descripción
Word16 saturate(Word32 L_var1)	Límite a 16 bits
Word16 add(Word16 var1, Word16 var2)	Adición corta
Word16 sub(Word16 var1, Word16 var2)	Sustracción corta
Word16 abs_s(Word16 var1)	Valor absoluto corto
Word16 sh1(Word16 var1, Word16 var2)	Desplazamiento corto a la izquierda
Word16 shr(Word16 var1, Word16 var2)	Desplazamiento corto a la derecha
Word16 mult(Word16 var1, Word16 var2)	Multiplicación corta
Word32 L_mult(Word16 var1, Word16 var2)	Multiplicación larga
Word16 negate(Word16 var1)	Negación corta
Word16 extract_h(Word32 L_var1)	Extracción alta
Word16 extract_l(Word32 L_var1)	Extracción baja
Word16 round(Word32 L_var1)	Redondeo
Word32 L_mac(Word32 L_var3, Word16 var1, Word16 var2)	Multiplicar y acumular
Word32 L_msu(Word32 L_var3, Word16 var1, Word16 var2)	Multiplicar y sustraer
Word32 L_add(Word32 L_var1, Word32 L_var2)	Adición larga
Word32 L_sub(Word32 L_var1, Word32 L_var2)	Sustracción larga
Word32 L_negate(Word32 L_var1)	Negación larga
Word16 mult_r(Word16 var1, Word16 var2)	Multiplicación con redondeo
Word32 L_sh1(Word32 L_var1, Word16 var2)	Desplazamiento largo a la izquierda
Word32 L_shr(Word32 L_var1, Word16 var2)	Desplazamiento largo a la derecha
Word16 shr_r(Word16 var1, Word16 var2)	Desplazamiento a la derecha con redondeo
Word16 mac_r(Word32 L_var3, Word16 var1, Word16 var2)	Mac con redondeo
Word16 msu_r(Word32 L_var3, Word16 var1, Word16 var2)	Msu con redondeo
Word32 L_deposit_h(Word16 var1)	var 1 a 16 bits en parte MSB
Word32 L_deposit_l(Word16 var1)	var 1 a 16 bits en parte LSB
Word32 L_shr_r(Word32 L_var1, Word16 var2)	Desplazamiento largo a la derecha con redondeo
Word32 L_abs(Word32 L_var1)	Valor absoluto largo
Word16 norm_s(Word16 var1)	Norma corta
Word16 div_s(Word16 var1, Word16 var2)	División corta
Word16 norm_l(Word32 L_var1)	Norma larga

CUADRO 12/G.729

Resumen de tablas en tab. ld8.c

Tabla	Tamaño	Descripción
tab_hup_s	28	Filtro de muestreo ascendente para postfiltro
tab_hup_1	112	Filtro de muestreo ascendente para postfiltro
inter_3	13	Filtro FIR para interpolar la correlación
inter_3	31	Filtro FIR para interpolar la excitación pasada
lspcb1	128 × 10	Cuantificador LSP (primera fase)
lspcb2	32 × 10	Cuantificador LSP (segunda fase)
fg	2 × 4 × 10	Predictores MA en el cuantificador vectorial LSP
fg_sum	2 × 10	Utilizado en el cuantificador vectorial LSP
fg_sum_inv	2 × 10	Utilizado en el cuantificador vectorial LSP
gbk1	8 × 2	Tabla de códigos GA en el cuantificador vectorial de ganancia
gbk2	16 × 2	Tabla de códigos GB en el cuantificador vectorial de ganancia
map1	8	Utilizado en el cuantificador vectorial de ganancia
imap1	8	Utilizado en el cuantificador vectorial de ganancia
map2	16	Utilizado en el cuantificador vectorial de ganancia
ima21	16	Utilizado en el cuantificador vectorial de ganancia
window	240	Ventana de análisis LP
lag_h	10	Ventana de retardo para expansión de anchura de banda (parte superior)
lag_l	10	Ventana de retardo para expansión de anchura de banda (parte inferior)
grid	61	Puntos de rejilla en la conversión de LP a LSP
tabsqr	49	Tabla de búsqueda en el cálculo de la raíz cuadrada inversa
tablog	33	Tabla de búsqueda en el cálculo de logaritmos de base 2
table	65	Tabla de búsqueda en la conversión de LSF a LSP y viceversa
slope	64	Pendientes lineales en la conversión de LSP a LSF
tabpow	33	Tabla de búsqueda en el cálculo de 2 ^x

CUADRO 13/G.729

Resumen de subprogramas específicos del codificador

Nombre de fichero	Descripción
acelp_co.c	Buscar tabla de códigos fijos
cod_ld8k.c	Subprograma del codificador
lpc.c	Análisis LP
pitch.c	Búsqueda de tono
pre_proc.c	Preprocesamiento (filtrado paso alto y escalamiento)
pwf.c	Cálculo de coeficientes de ponderación perceptual
qua_gain.c	Cuantificador de ganancia
qua_lsp.c	Cuantificador LSP

CUADRO 14/G.729

Resumen de subprogramas específicos del decodificador

Nombre de fichero	Descripción
de_acelp.c	Decodificar tabla de códigos algebraicos
dec_gain.c	Decodificar ganancias
dec_lag3.c	Decodificar el índice de tabla de códigos adaptables
dec_ld8k.c	Subprograma del decodificador
lspdec.c	Subprograma de decodificación LSP
post_pro.c	Postprocesamiento (filtrado paso alto y escalamiento)
pst.c	Subprogramas de postfiltro

CUADRO 15/G.729

Resumen de subprogramas generales

Nombre de fichero	Descripción
basicop2.c	Operadores básicos
oper_32b.c	Operadores básicos ampliados
bits.c	Subprogramas de manipulación de bits
dspfunc.c	Funciones matemáticas
filter.c	Funciones de filtro
gainpred.c	Predictor de ganancia
lpcfunc.c	Subprogramas varios relativos al filtro LP
lspgetq.c	Cuantificador LSP
p_parity.c	Cálculo de paridad de tono
pred_1t3.c	Generación de tabla de códigos adaptativos
util.c	Funciones de utilidad