

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

G.729.1

Amendment 6

(03/2010)

SERIES G: TRANSMISSION SYSTEMS AND MEDIA,
DIGITAL SYSTEMS AND NETWORKS

Digital terminal equipments – Coding of voice and audio
signals

G.729-based embedded variable bit-rate coder: An
8-32 kbit/s scalable wideband coder bitstream
interoperable with G.729

**Amendment 6: New Annex E on superwideband
scalable extension**

Recommendation ITU-T G.729.1 (2006) –
Amendment 6

ITU-T G-SERIES RECOMMENDATIONS
TRANSMISSION SYSTEMS AND MEDIA, DIGITAL SYSTEMS AND NETWORKS

INTERNATIONAL TELEPHONE CONNECTIONS AND CIRCUITS	G.100–G.199
GENERAL CHARACTERISTICS COMMON TO ALL ANALOGUE CARRIER-TRANSMISSION SYSTEMS	G.200–G.299
INDIVIDUAL CHARACTERISTICS OF INTERNATIONAL CARRIER TELEPHONE SYSTEMS ON METALLIC LINES	G.300–G.399
GENERAL CHARACTERISTICS OF INTERNATIONAL CARRIER TELEPHONE SYSTEMS ON RADIO-RELAY OR SATELLITE LINKS AND INTERCONNECTION WITH METALLIC LINES	G.400–G.449
COORDINATION OF RADIOTELEPHONY AND LINE TELEPHONY	G.450–G.499
TRANSMISSION MEDIA AND OPTICAL SYSTEMS CHARACTERISTICS	G.600–G.699
DIGITAL TERMINAL EQUIPMENTS	G.700–G.799
General	G.700–G.709
Coding of voice and audio signals	G.710–G.729
Principal characteristics of primary multiplex equipment	G.730–G.739
Principal characteristics of second order multiplex equipment	G.740–G.749
Principal characteristics of higher order multiplex equipment	G.750–G.759
Principal characteristics of transcoder and digital multiplication equipment	G.760–G.769
Operations, administration and maintenance features of transmission equipment	G.770–G.779
Principal characteristics of multiplexing equipment for the synchronous digital hierarchy	G.780–G.789
Other terminal equipment	G.790–G.799
DIGITAL NETWORKS	G.800–G.899
DIGITAL SECTIONS AND DIGITAL LINE SYSTEM	G.900–G.999
MULTIMEDIA QUALITY OF SERVICE AND PERFORMANCE – GENERIC AND USER-RELATED ASPECTS	G.1000–G.1999
TRANSMISSION MEDIA CHARACTERISTICS	G.6000–G.6999
DATA OVER TRANSPORT – GENERIC ASPECTS	G.7000–G.7999
PACKET OVER TRANSPORT ASPECTS	G.8000–G.8999
ACCESS NETWORKS	G.9000–G.9999

For further details, please refer to the list of ITU-T Recommendations.

Copyright © 2010 ITU-T. All rights reserved.

Recommendation ITU-T G.729.1

G.729-based embedded variable bit-rate coder: An 8-32 kbit/s scalable wideband coder bitstream interoperable with G.729

Amendment 6

New Annex E on superwideband scalable extension

Summary

Recommendation ITU-T G.729.1 describes an 8-32 kbit/s scalable wideband speech and audio coding algorithm interoperable with ITU-T G.729, ITU-T G.729A and ITU-T G.729B.

The output of the ITU-T G.729EV coder has a bandwidth of 50-4000 Hz at 8 and 12 kbit/s and 50-7000 Hz from 14 to 32 kbit/s. At 8 kbit/s, ITU-T G.729EV is fully interoperable with ITU-T G.729, Annex A/G.729 and Annex B/G.729. Hence, an efficient deployment in existing ITU-T G.729-based VoIP infrastructures is foreseen. The coder operates on 20 ms frames and has an algorithmic delay of 48.9375 ms. By default, the encoder input and decoder output are sampled at 16 kHz.

The encoder produces an embedded bitstream structured in 12 layers corresponding to 12 available bit rates from 8 to 32 kbit/s. The bitstream can be truncated at the decoder side or by any component of the communication system to adjust "on the fly" the bit rate to the desired value with no need for outband signalling.

The underlying algorithm is based on a three-stage coding structure: embedded Code-Excited Linear Prediction (CELP) coding of the lower band (50-4000 Hz), parametric coding of the higher band (4000-7000 Hz) by Time-Domain Bandwidth Extension (TDBWE), and enhancement of the full band (50-7000 Hz) by a predictive transform coding technique referred to as Time-Domain Aliasing Cancellation (TDAC).

Amendment 1 introduces the new Annex A containing the RTP payload format, capability identifiers and parameters for signalling of ITU-T G.729.1 capabilities using ITU-T H.245. Both format and capability parameters are fully compatible with the corresponding ITU-T G.729.1 RTP definitions to allow seamless interoperability. Besides the new Annex, Amendment 1 to ITU-T G.729.1 incorporates changes needed to correct defects in ITU-T G.729.1 and provides new, more comprehensive test vectors.

Amendment 2 introduces the new Annex B, which defines an alternative implementation of the ITU-T G.729.1 algorithm using floating point arithmetic to be used for implementation on DSP hardware optimized for floating-point operations. The accompanying floating point C-code is fully interoperable with the fixed-point C-code.

Amendment 3 extends the low-delay functionality of main body and Annex B to the first wideband bit rate (14 kbit/s). It also incorporates changes needed to correct defects in the text and C-code of ITU-T G.729.1 main body and Annex B.

Amendment 4 introduces a new Annex C specifying a discontinuous transmission (DTX) and comfort noise generation for ITU-T G.729.1. With this annex, the ITU-T G.729.1 encoder is capable of generating a silence insertion description (SID) each time an update of the ambient background noise parameters is required to maintain the quality of the generated background noise. The SID information includes a core lower band layer, which can be decoded by the decoder of Annex B/G.729, an enhancement lower band layer and a higher band layer. The non-transmission



Documents

Document Sharing Hub

between SID updates and the small size of the SID provide a significant reduction of bandwidth during inactive segments. Besides this new annex, Amendment 4 incorporates changes needed to correct defects identified in ITU-T G.729.1 C source code (main body and Annex B), provides a revised set of test vectors, and updates the complexity figures table of ITU-T G.729.1 text.

Annex D introduced by Amendment 5 provides an alternative implementation using floating point arithmetic of the discontinuous transmission (DTX) and comfort noise generation (CNG) of Annex C – which uses fixed-point arithmetic. Besides this new annex, Amendment 5 incorporates changes needed to correct defects identified in the ITU-T G.729.1 C source code for its main body and Annex B, and provides a revised set of test vectors.

Corrigendum 1 addressed some problems discovered recently in the ANSI C-codes of the main body of ITU-T G.729.1 and of its Annexes B, C and D, in the so-called Release 1.5 of the code.

Amendment 6 brings in new Annex E a scalable superwideband (SWB, 50-14000 Hz) speech and audio coding algorithm operating from 36 to 64 kbit/s and interoperable with ITU-T G.729 and ITU-T G.729.1.

For consistency, the existing ANSI-C code for the whole of ITU-T G.729.1 is reissued as part of this publication and labelled as *Release 1.6*, without any additional change. Test vectors that complement this release are also available in the ITU-T test signal database at <http://itu.int/net/ITU-T/sigdb/speaudio/Gseries.htm#ITU-T G.729.1>.

History

Edition	Recommendation	Approval	Study Group
1.0	ITU-T G.729.1	2006-05-29	16
1.1	ITU-T G.729.1 (2006) Amend. 1	2007-01-13	16
1.2	ITU-T G.729.1 (2006) Amend. 2	2007-02-13	16
1.3	ITU-T G.729.1 (2006) Amend. 3	2007-08-29	16
1.4	ITU-T G.729.1 (2006) Amend. 4	2008-06-29	16
1.5	ITU-T G.729.1 (2006) Amend. 5	2008-12-07	16
1.6	ITU-T G.729.1 (2006) Cor.1	2009-08-29	16
1.7	ITU-T G.729.1 (2006) Amend.6	2010-03-29	16

FOREWORD

The International Telecommunication Union (ITU) is the United Nations specialized agency in the field of telecommunications, information and communication technologies (ICTs). The ITU Telecommunication Standardization Sector (ITU-T) is a permanent organ of ITU. ITU-T is responsible for studying technical, operating and tariff questions and issuing Recommendations on them with a view to standardizing telecommunications on a worldwide basis.

The World Telecommunication Standardization Assembly (WTSA), which meets every four years, establishes the topics for study by the ITU-T study groups which, in turn, produce Recommendations on these topics.

The approval of ITU-T Recommendations is covered by the procedure laid down in WTSA Resolution 1.

In some areas of information technology which fall within ITU-T's purview, the necessary standards are prepared on a collaborative basis with ISO and IEC.

NOTE

In this Recommendation, the expression "Administration" is used for conciseness to indicate both a telecommunication administration and a recognized operating agency.

Compliance with this Recommendation is voluntary. However, the Recommendation may contain certain mandatory provisions (to ensure e.g., interoperability or applicability) and compliance with the Recommendation is achieved when all of these mandatory provisions are met. The words "shall" or some other obligatory language such as "must" and the negative equivalents are used to express requirements. The use of such words does not suggest that compliance with the Recommendation is required of any party.

INTELLECTUAL PROPERTY RIGHTS

ITU draws attention to the possibility that the practice or implementation of this Recommendation may involve the use of a claimed Intellectual Property Right. ITU takes no position concerning the evidence, validity or applicability of claimed Intellectual Property Rights, whether asserted by ITU members or others outside of the Recommendation development process.

As of the date of approval of this Recommendation, ITU had received notice of intellectual property, protected by patents, which may be required to implement this Recommendation. However, implementers are cautioned that this may not represent the latest information and are therefore strongly urged to consult the TSB patent database at <http://www.itu.int/ITU-T/ipr/>.

© ITU 2010

All rights reserved. No part of this publication may be reproduced, by any means whatsoever, without the prior written permission of ITU.

CONTENTS

	Page
Amendment 6 – New Annex E on superwideband scalable extension.....	1
E.1 Scope	1
E.2 Overview	1
E.3 Abbreviations	1
E.4 Mathematical expressions	2
E.5 General description of the coder.....	3
E.6 Functional description of the encoder	4
E.7 Functional description of the decoder	37
E.8 Description of the transmitted parameter indices	66
E.9 Bit-exact description of the SWB extension	68
Bibliography.....	69
Electronic attachment: ANSI C source code	

Recommendation ITU-T G.729.1

G.729-based embedded variable bit-rate coder: An 8-32 kbit/s scalable wideband coder bitstream interoperable with G.729

Amendment 6

New Annex E on superwideband scalable extension

(This annex forms an integral part of this Recommendation)

E.1 Scope

This annex contains the description of a scalable superwideband (SWB, 50-14000 Hz) speech and audio coding algorithm operating from 36 to 64 kbit/s and interoperable with ITU-T G.729 and ITU-T G.729.1.

Annex E is organized as follows. The abbreviations and mathematical expressions used throughout this annex are defined in clauses E.3 and E.4, respectively. Clause E.5 gives a general outline of the algorithm. The encoder and decoder principles are discussed in clauses E.6 and E.7, respectively. The transmitted parameters are presented in clause E.8. Clause E.9 describes the software that defines this coder in 16-32 bits fixed-point arithmetic.

E.2 Overview

The output of the ITU-T G.729.1 SWB coder has a bandwidth of 50-14000 Hz. The coder operates on 20 ms frames and has an algorithmic delay of 55.6875 ms. By default, the encoder input and decoder output are sampled at 32 kHz.

The superwideband encoder produces an embedded bitstream structured in five layers corresponding to five available bit rates from 36 to 64 kbit/s in addition to the twelve layers of ITU-T G.729.1. The bitstream can be truncated at the decoder side or by any component of the communication system to instantaneously adjust the bit rate to the desired value with no need for out-of-band signalling. At 32 kbit/s, ITU-T G.729.1 SWB is fully interoperable with ITU-T G.729.1.

The underlying algorithm is based on a two-stage coding structure: bandwidth extension coding in the modified discrete cosine transform (MDCT) domain of band 7000-14000 Hz, and enhancement of MDCT coding in the band 50-7000 Hz by vector quantization of the MDCT error.

Annex E contains an electronic attachment with the ANSI C source code, which is an integral part of this annex, as well as a set of test vectors. The latter can be found online in the ITU-T test signal database at <http://www.itu.int/net/ITU-T/sigdb/speaudio/Gseries.htm#ITU-T G.729.1>.

E.3 Abbreviations

The acronyms used in this annex are summarized in Table E.1.

Table E.1 – Glossary of acronyms

Acronym	Description
ACELP	Algebraic CELP
CELP	Code-Excited Linear Prediction
DFT	Discrete Fourier Transform
FEC	Frame Erasure Concealment
GLCVQ	Gosset Low Complexity Vector Quantization
HF	High Frequency (7-14 kHz)
IDFT	Inverse Discrete Fourier Transform
IIR	Infinite Impulse Response
iMDCT	Inverse MDCT
LF	Low Frequency (0-7 kHz)
MDCT	Modified Discrete Cosine Transform
NB	Narrowband
SVQ	Spherical Vector Quantization
SWB	Superwideband
TB	Transition Band
VQ	Vector Quantization
WB	Wideband
WMOPS	Weighted Million Operations Per Second

E.4 Mathematical expressions

The following conventions apply to mathematical expressions in this annex:

- $\lfloor x \rfloor$ indicates the largest integer less than or equal to x : $\lfloor 1.1 \rfloor = 1$, $\lfloor 1.0 \rfloor = 1$, and $\lfloor -1.1 \rfloor = -2$;
- $\lceil x \rceil$ indicates the smallest integer greater than or equal to x : $\lceil 1.1 \rceil = 2$, $\lceil 2.0 \rceil = 2$, and $\lceil -1.1 \rceil = -1$;
- $|x|$ indicates the absolute value of x : $|17| = 17$, $|-17| = 17$;
- $\min(x_0, x_1, \dots, x_{N-1})$ indicates the minimum of $x_0, x_1, \dots, x_{N-1}$, N being the number of components;
- $\max(x_0, x_1, \dots, x_{N-1})$ indicates the maximum of $x_0, x_1, \dots, x_{N-1}$;
- $\text{sgn}(x) = \begin{cases} 1, & \text{if } x \geq 0, \\ -1, & \text{otherwise} \end{cases}$
- $\mathbf{A}^T$ indicates the transpose of matrix $\mathbf{A}$;
- $x \bmod y$ indicates the remainder after dividing x by y : $x \bmod y = x - (y \lfloor x / y \rfloor)$;
- $\text{round}(x)$ is traditional rounding: $\text{round}(x) = \text{sgn}(x) \cdot \lfloor |x| + 0.5 \rfloor$;
- $\exp(x)$ is equivalent to e^x where e is the base of the natural algorithm;
- $\sum$ indicates summation;
- $\prod$ indicates product;

- $\wedge$ indicates logical conjunction or AND operation;
- $\vee$ indicates logical disjunction or OR operation;
- $\text{Re}(z)$ indicates the real part of complex variable z ;
- $\text{Im}(z)$ indicates the imaginary part of complex variable z .

Unless otherwise specified, $\log(x)$ denotes logarithm at base 10, throughout this annex.

E.5 General description of the coder

The ITU-T G.729.1 superwideband extension comprises 5 layers extending the wideband coding of ITU-T G.729.1 to an embedded superwideband coding, with total bit rates of 36, 40, 48, 56, and 64 kbit/s. The first SWB extension layer comprises a two-mode coding of the high frequencies using Generic mode or Sinusoidal mode coding. In the Generic mode, the coding is based on adaptive replication of coded wideband signal sub-bands. The Sinusoidal mode coding adds sinusoids to the high frequency content. The mode selection depends on the input signal characteristics. The second extension layer implements an adaptive bit allocation between low-frequency and high-frequency signal improvements, while the third layer further improves the high-frequency quality by means of additional sinusoidal components. The fourth and fifth extension layers provide wideband signal improvement.

Table E.2 summarizes the extension layer technologies and bit rates.

Table E.2 – Layer structure of ITU-T G.729.1 superwideband extension

Layer	Bit rate	Technology
ITU-T G.729.1 codec	32 kbit/s	Embedded NB/WB codec comprising 12 layers
Layer 6mo	+4 kbit/s	Generic mode/Sinusoidal mode HF coding
Layer 7mo	+4 kbit/s	Additional HF sinusoids/Wideband improvement
Layer 8mo	+8 kbit/s	Additional HF sinusoids
Layer 9mo	+8 kbit/s	Wideband improvement
Layer 10mo	+8 kbit/s	Wideband improvement

E.5.1 Input/output sampling rate

The ITU-T G.729.1 SWB extension accepts signals sampled at 32 kHz for SWB coding, and wideband (WB) or narrowband (NB) signals sampled at 16 and 8 kHz, respectively, for ITU-T G.729.1 operation, where the SWB extension is bypassed. Similarly, the decoder output can be sampled at 32, 16, or 8 kHz, with SWB rendering at 32 kHz.

E.5.2 Algorithmic delay

The input signal is processed using a 20-ms frame size, corresponding with the ITU-T G.729.1 WB coding. The additional algorithmic delay due to the signal sampling rate conversion operations and post-processing is 6.75 ms.

E.5.3 Complexity and memory consumption

The worst case complexity of the ITU-T G.729.1 SWB encoding and decoding was computed using the test vector set and is 62.55 WMOPS in the bit-rate switching condition. Table E.3 summarizes the computational complexity of the extension with different bit rates including switching to ITU-T G.729.1 output (32 kbit/s) sampled at 32 kHz.

Table E.3 – Complexity (in WMOPS)

Option	Bit rate	Encoder	Decoder	Total
SWB	32 kbit/s	22.31	20.36 (Note)	42.67
	36 kbit/s	33.61	22.35 (Note)	55.96
	40 kbit/s	34.36	22.51 (Note)	56.87
	48 kbit/s	34.95	19.14 (Note)	54.09
	56 kbit/s	38.70	20.49	59.19
	64 kbit/s	38.67	21.06 (Note)	59.73
Bit-rate switching	–	38.72	24.55	63.27
NOTE – Calculated with 15% frame erasure rate.				

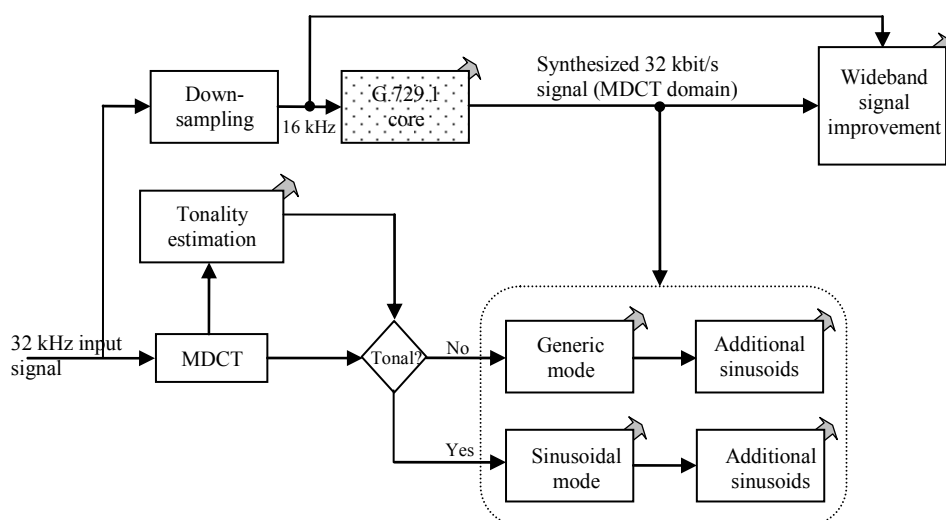
Table E.4 presents the memory consumption of the SWB extension. The total table ROM consumption is about 13.2 kWords and the total static RAM usage is about 5.7 kWords. The worst case for the dynamic RAM between the encoder and the decoder is about 9.9 kWords.

Table E.4 – Memory

Memory	Encoder	Decoder	Common	Total
ROM (kWord)	–	4.3	8.9	13.2
RAM (kWord)	1.6	4.1	–	5.7

E.6 Functional description of the encoder

The structure of the encoder is presented in Figure E.1. From the figure it can be seen that while the ITU-T G.729.1 core codec operates on a 16 kHz signal, the SWB extension uses a 32 kHz signal. The SWB encoding is performed in the MDCT domain. Two alternative modes, Generic mode and Sinusoidal mode, are used for coding the first layer of the extension [b-SWB]. The decision between these modes is done based on the estimated tonality of the input signal. Higher SWB layers are coded using either additional sinusoids, which improve the quality of the high frequency content, or wideband signal improvement, which is used to improve the perceptual quality of wideband content.



Note: Grey arrows indicate inputs to the bitstream.

Figure E.1 – Structural block diagram of the encoder

E.6.1 Sampling conversion

For the ITU-T G.729.1 core coding, the 32 kHz input signal is downsampled to 16 kHz. The conversion is performed for each 20 ms input frame. First, the input signal $s_{inp}(n)$ is high-pass filtered using a 2nd order high-pass filter with cut-off frequency at 50 Hz in order to suppress the low frequency components. The resulting signal $s_{32}(n)$ is then downsampled using a 2-stage infinite impulse response (IIR) filter as given for the current frame by the first-stage equation:

$$s_{stage1}(n) = \sum_{i=0}^5 b_{i,stage1} s_{32}(n-i) - \sum_{j=1}^5 a_{j,stage1} s_{stage1}(n-j) \quad n = 0, \dots, 639 \quad (E-1)$$

and the second-stage equation:

$$s_{stage2}(n) = \sum_{i=0}^4 b_{i,stage2} s_{stage1}(n-i) - \sum_{j=1}^4 a_{j,stage2} s_{stage2}(n-j) \quad n = 0, \dots, 639 \quad (E-2)$$

where the negative indices of the processed signal correspond to previous frame memory. The filter coefficients $b_{k,stage1}$ and $a_{k,stage1}$ are given in Table E.5.

Finally, the downsampled signal $s_{16}(n)$ is obtained as

$$s_{16}(n) = s_{stage2}(k) \quad n = 0, \dots, 319 \quad k = 0, 2, \dots, 638 \quad (E-3)$$

Table E.5 – Downsampling filter coefficients

$b_{k,stage1}$	$a_{k,stage1}$	$b_{k,stage2}$	$a_{k,stage2}$
0.0581908	1.0	0.0708131	1.0
0.1273200	−1.462269006103	0.1446910	−1.308055753351
0.2007510	2.194832312468	0.1941500	1.509406808301
0.2007510	−1.638502434011	0.1446910	−0.8377333289311
0.1273200	0.926185061962	0.0708131	0.2783957005103
0.0581908	−0.2701781844527		

E.6.2 WB encoding

The WB coding is performed for the downsampled input signal $s_{16}(n)$ as defined here, in Rec. ITU-T G.729.1.

E.6.3 Transformation to MDCT domain

The input 32 kHz signal is transformed to MDCT domain. The window must be synchronized with the MDCT transform of the WB codec, i.e., the input frame is delayed such that the delay caused by downsampling to 16 kHz as well as the delay caused by core coding in the ITU-T G.729.1 encoder are considered.

The transform length is $N=1280$ points for 32 kHz superwideband signal, composed of 640 samples from the current frame, together with the $M=640$ ($= N/2$) samples from the previous frame:

$$x(n) = s'_{32}(n - \frac{N}{2}) \quad n = 0, \dots, N-1 \quad (E-4)$$

where $s'_{32}(n)$ is the delayed signal $s_{32}(n)$ to take into account the ITU-T G.729.1 core encoding delay.

The forward transform can be formulated by the following equations:

$$M_{32}(2k) = A \cdot \text{Re} \left\{ b \cdot W_N^{k+0.5} \sum_{n=0}^{N/4-1} u(n) \cdot a \cdot W_N^{n+0.5} W_{N/4}^{nk} \right\} \quad k = 0, \dots, N/4 - 1 \quad (\text{E-5})$$

$$M_{32}(N/2 - 1 - 2k) = A \cdot \text{Im} \left\{ b \cdot W_N^{k+0.5} \sum_{n=0}^{N/4-1} u(n) \cdot a \cdot W_N^{n+0.5} W_{N/4}^{nk} \right\} \quad k = 0, \dots, N/4 - 1 \quad (\text{E-6})$$

where $W_N = e^{-j\frac{2\pi}{N}}$ $= \cos(\frac{2\pi}{N}) - j \sin(\frac{2\pi}{N})$, and $u(n)$ are the samples after windowing and twiddling.

The forward transform can be divided into four steps:

1) Pre-processing

Pre-processing includes windowing and twiddling to obtain the pre-processed signal $u(n)$.

$$y(n) = h(n)x(n) \quad n = 0, \dots, N - 1 \quad (\text{E-7})$$

where $h(n)$ is the analysis window. If the synthesis window is defined as $g(n)$, the analysis window and the synthesis window must satisfy the perfect reconstruction condition:

$$h(n)g(n) + h(n + N/2)g(n + N/2) = 1, \quad n = 0, \dots, N - N/2 - 1 \quad (\text{E-8})$$

An asymmetric window shape is used to reduce the delay associated to the transform coding stage by 10 ms while keeping the number of frequency coefficients unchanged. As opposed to traditional MDCT windows, this window is not symmetric; the second half of the window being different from the time reversed version of the first half.

The original analysis asymmetric window shape was designed for 16 kHz input signal and comes from clause 6.11.2 of [b-ITU-T G.718] as given by the following equation:

$$w_a(n) = \frac{w_i(n)}{\sqrt{D(n)}} \quad 0 \leq n < M \quad (\text{E-9})$$

where

$$w_i(n) = \begin{cases} \sin \left[\left(n + \frac{1}{2} \right) \frac{\pi}{\left(M - \frac{M}{8} \right)} \right] & 0 \leq n < M - \frac{M}{8} \\ 0, & M - \frac{M}{8} \leq n < M \end{cases} \quad (\text{E-10})$$

and $D(n)$ is defined by

$$D(n) = w_i(n)w_i(M - 1 - n) + w_i\left(n + \frac{M}{2}\right)w_i\left(\frac{M}{2} - 1 - n\right) \quad 0 \leq n < \frac{M}{2} \quad \text{and} \quad (\text{E-11})$$

$$D\left(n + \frac{M}{2}\right) = D(n) \quad 0 \leq n < \frac{M}{2} \quad (\text{E-12})$$

where $M/8$ is the number of trailing zeros.

To deal with an input signal sampled at 32 kHz, this window is updated and interpolated such as to define an MDCT transform of size M instead of size $M/2$. Hence, the N samples of the window are stored in two separated arrays. The first one contains the M taps obtained directly from the asymmetric window $w_a(n)$, as done in clause 6.11.2 of [b-ITU-T G.718]. The second one contains the M interpolated window taps. The interpolation process inserts a coefficient between each coefficient of the initial window $w_a(n)$ defining a new window having twice the initial length.

The interpolated window is built as follows:

The even terms use directly the taps of the asymmetric window from clause 6.11.2 of [b-ITU-T G.718]:

$$\begin{cases} h(2n) = w_a(n) \\ h(M+2n) = w_a(M/2+n) \end{cases} \quad 0 \leq n < M/2 \quad (\text{E-13})$$

And the odd terms are linearly interpolated:

$$\begin{cases} h(2n+1) = P_n \cdot [w_a(n) + w_a(n+1)] \\ h(M+2n+1) = P_n \cdot [w_a(M/2+n) + w_a(M/2+n+1)] \end{cases} \quad (\text{E-14})$$

with $w_a(M)=0$, and with the correction factor P_n ensuring the perfect reconstruction for the MDCT stage:

$$P_n = [1 + w_a(n+1) \cdot w_a(M-1-n) + w_a(M/2+n+1) \cdot w_a(M/2-1-n)]^{-1} \quad (\text{E-15})$$

For this updated window of size $N = 2M$, the final amount of trailing zeros is $M_4 = N/8$.

After the windowing, the elements of $y(n)$ are twiddled via the following operations:

$$\begin{cases} z(n + N/4) = y(n) - y(N/2 - 1 - n) \\ z(N/4 - 1 - n) = -y(N - 1 - n) - y(N/2 + n) \end{cases} \quad n = 0, \dots, N/4 - 1 \quad (\text{E-16})$$

and

$$u(n) = z(2n) + jz(N/2 - 1 - 2n) \quad n = 0, \dots, N/4 - 1 \quad (\text{E-17})$$

The data after pre-processing is $u(n)$, it is a complex number, the real part is $z(2n)$, and the imaginary part is $z(N/2 - 1 - 2n)$.

2) Pre-rotation

The rotation is conducted to twiddled data:

$$v(n) = u(n) \cdot a \cdot W_N^{n+0.5} \quad n = 0, \dots, N/4 - 1 \quad (\text{E-18})$$

where a is the scalar to the rotation factor $W_N^{n+0.5}$, and $W_N = e^{-j\frac{2\pi}{N}} = \cos(\frac{2\pi}{N}) - j\sin(\frac{2\pi}{N})$. To reduce the complexity, the scalar is integrated into the rotation factor, the specific value in this application is $a = \frac{\sqrt{2}}{\sqrt[4]{N}}$. Since the real and the imaginary parts of the rotation factor $W_N^{n+0.5}$ satisfy the following condition:

$$\cos(\frac{2\pi}{N}(n+0.5)) = \sin(\frac{2\pi}{N}(L-1-n+0.5)) \quad (\text{E-19})$$

the real and the imaginary parts of the rotation factor with a scalar are equal, and only the real parts (or imaginary parts) $\frac{1}{4\sqrt{320}} \cos(\frac{2\pi}{N}(n+0.5))$, $n = 0, \dots, 319$ are stored as a table with $N/4$ points.

3) Discrete Fourier transform (DFT)

A complex DFT with length L ($L = N/4 = 320$) is applied to the rotated data $v(n)$:

$$r(k) = \sum_{n=0}^{L-1} v(n) W_L^{nk} \quad k = 0, \dots, L-1 \quad (\text{E-20})$$

Here, a simple power-2 DFT is not suitable, so it is implemented with the following low complexity 2-dimensional ($L = N/4 = P \times Q$) DFT, where $P=64$ and $Q=5$ are coprime factors.

To reduce complexity, an address table is introduced. It can be calculated by:

$$I(n_1, n_2) = (K_1 \times n_1 + K_2 \times n_2) \bmod L \quad \begin{array}{l} n_1 = 0, \dots, P-1 \\ n_2 = 0, \dots, Q-1 \end{array} \quad (\text{E-21})$$

Where K_1, K_2 are coprime and satisfy the condition $(K_1 K_2) \bmod L = 1$.

Here, $K_1 = 65$, $K_2 = 256$. The address table I is stored for low complexity 2-dimensional DFT, and is used to indicate which samples are used for P-point DFT or Q-point DFT following:

a) Applying P-point DFT to $v(n)$ for Q times based on the address table I .

The input data to the i -th ($i=0, \dots, Q-1$) P-point DFT is found by seeking their addresses stored in the address table I . For the i -th P-point DFT, the addresses of the input data are the P continuous elements starting from the $i * P$'s element in table I . For every time of P-point DFT, the resulting data need to be applied a circular shift with a step of x , where x is the re-ordered index, which satisfies $\left(x \cdot \left(\left(\frac{K_1^2}{Q} \right) \bmod P \right) \right) \bmod P = 1$

The output of step a) is:

$$w(k) = DFT_P(v(I + iP))_x \quad i = 0, \dots, Q-1 \quad (\text{E-22})$$

For the i -th ($i=0, \dots, 4$) 64-point DFT, the addresses of the input data are the 64 continuous elements starting from $I[64i]$, and the results are circular shifted with $x=5$. Here is an example of circular shift, the original vector is $Z = [z_0 \ z_1 \ z_2 \ z_3 \ z_4]$, the new vector with 2 circular shift is $Z = [z_0 \ z_2 \ z_4 \ z_1 \ z_3]$.

b) Applying Q-point DFT to $w(k)$ for P times based on the address table I .

The input data to the i -th ($i=0, \dots, P-1$) Q-point DFT is found by seeking their addresses stored in the address table I . For the i -th Q-point DFT, the addresses of the input data are the Q elements starting from the i -th element in table I , each of which separated by a step of P . For every time of Q-point DFT, the resulting data need to be applied a circular shift with a step of y . y is the re-ordered index, which satisfies $\left(y \cdot \left(\left(\frac{K_2^2}{P} \right) \bmod Q \right) \right) \bmod Q = 1$.

The output of step b) is:

$$r(k) = DFT_Q(w(I + iP))_y \quad i = 0, \dots, P-1 \quad (\text{E-23})$$

For the i -th ($i=0,\dots,63$) 5-point DFT, the addresses of the input data are the 5 continuous elements starting from $I[i]$, each of which is separated by a step of 64, and the results are circular shifted with $y=4$.

4) Post-rotation and output the spectrum coefficients

Post-rotation is conducted to the samples after DFT $r(k)$:

$$l(k) = b \cdot W_N^{k+0.5} r(k) \quad k = 0, \dots, N/4 - 1 \quad (\text{E-24})$$

where b is the scalar to the rotation factor. The specific value in this application is $b = \frac{\sqrt{2}}{\sqrt[4]{N}}$.

Finally, the spectrum coefficients are:

$$M_{32}(2k) = A \cdot \text{Re}\{l(k)\} \quad k = 0, \dots, N/4 - 1 \quad (\text{E-25})$$

$$M_{32}(N/2 - 1 - 2k) = A \cdot -\text{Im}\{l(k)\} \quad k = 0, \dots, N/4 - 1 \quad (\text{E-26})$$

where A is the normalization factor, and it is set to 1 for simplicity.

E.6.4 Tonality estimation

The coding mode of the first SWB layer, Layer 6mo, is selected based on the tonality estimation, which is performed in the MDCT domain by comparing the logarithmic domain energies of the current and previous frame.

The tonality estimation is based on a correlation analysis between the spectral peaks in the current frame and the past frame. The input to the algorithm is the log-energy MDCT spectrum in the HF frequency range, defined as:

$$E_{dB}(k - 280) = 10 \log[M_{32}^2(k)] \quad k = 280, \dots, 559 \quad (\text{E-27})$$

where $M_{32}(k)$ is defined in equations E-25 and E-26. In this clause, the term "spectrum" will refer to the log-energy MDCT spectrum, as defined by the above equation. The algorithm proceeds in four stages.

1) Spectral floor calculation

In the first stage, spectral floor is calculated and subtracted from the spectrum. This is done by applying a moving-average (MA) filter. The length of the filter is 31 samples and was found experimentally. The first value of the spectral floor is calculated at a position $i = 15$ which is a central bin of the first 31 samples. In the following text, $L_{MA} = 15$ is half of the filter length. That is:

$$E_{fl}(L_{MA}) = \frac{1}{2L_{MA} + 1} \sum_{i=0}^{2L_{MA}} E_{dB}(i) \quad (\text{E-28})$$

The next samples of the spectral floor are calculated recursively as

$$E_{fl}(k) = E_{fl}(k - 1) + \frac{1}{2L_{MA} + 1} (E_{dB}(k + L_{MA}) - E_{dB}(k - L_{MA} - 1)), \quad k = L_{MA} + 1, \dots, 279 - L_{MA} \quad (\text{E-29})$$

The missing parts, i.e., the first and the last L_{MA} samples are extrapolated as follows:

$$\begin{aligned} E_{fl}(k) &= 0.9E_{fl}(k + 1) + 0.1E_{dB}(k) & k = L_{MA} - 1, \dots, 0 \\ E_{fl}(k) &= 0.9E_{fl}(k - 1) + 0.1E_{dB}(k) & k = 280 - L_{MA}, \dots, 279 \end{aligned} \quad (\text{E-30})$$

Thus, we have 280 samples of the spectral floor which are subtracted from the original spectrum:

$$E_{res}(k) = E_{dB}(k) - E_{fl}(k) \quad k = 0, \dots, 279 \quad (E-31)$$

2) Amplitude smoothing and frequency compression

After subtracting the spectral floor, the spectrum (amplitudes) is smoothed and compressed by two in the X-axis (frequency bins) using the following method. The compression takes place in the leading and the trailing part of the spectrum. The central part is just smoothed. The leading part occupies a frequency range delimited by bins from 1 to 44 in the compressed domain. The frequency bin compression and amplitude smoothing are carried out in one operation as follows:

$$X_{dB}(k) = 0.3[0.65E_{res}(2k-2) + E_{res}(2k-1) + E_{res}(2k) + 0.65E_{res}(2k+1)] \quad k = 1, \dots, 44 \quad (E-32)$$

The central part comprising a range of bins from 45 to 56 is not compressed. It is only smoothed as:

$$X_{dB}(k) = 0.33[E_{res}(k+42) + E_{res}(k+43) + E_{res}(k+44)] \quad k = 45, \dots, 56 \quad (E-33)$$

The trailing part comprising a range of bins from 57 to 145 is calculated in the same way as the leading part. That is:

$$X_{dB}(k+57) = 0.3[0.65E_{res}(2k+99) + E_{res}(2k+100) + E_{res}(2k+101) + 0.65E_{res}(2k+102)] \\ k = 0, \dots, 88 \quad (E-34)$$

The sample with index 0 is copied from the uncompressed spectrum, i.e., $X_{dB}(0) = E_{res}(0)$. The sample with index 146 is calculated as follows:

$$X_{dB}(146) = 0.33[E_{res}(277) + E_{res}(278) + E_{res}(279)] \quad (E-35)$$

Thus, the total length of the compressed spectrum is $N_{tot} = 147$ bins. In case all bins of X_{dB} are negative, the following steps are not carried out and the final decision is declared as non-tonal.

3) Search of the local minima

In this stage, the indexes of local minima of the compressed spectrum are searched in a loop, described by the following formula and stored in a buffer i_{min} . That is:

$$i_{min} = (\forall i : (X_{dB}(i-1) > X_{dB}(i)) \wedge (X_{dB}(i) < X_{dB}(i+1))) \quad i = 1, \dots, 145 \quad (E-36)$$

The index 0 is added to i_{min} if $X_{dB}(0) < X_{dB}(1)$. Similarly, the index 146 is added to i_{min} if $X_{dB}(146) < X_{dB}(145)$. Let us denote the total number of minima found as N_{min} .

4) Calculation of the correlation map

The last stage is the calculation of a correlation map and a long-term correlation map. This is again a piece-wise operation. Thus, it is carried out on a peak-by-peak basis since the minima delimit the peaks. In the following text we will use the term "peak" to denote a section between two consecutive minima in the compressed spectrum X_{dB} .

Let us denote the compressed spectrum of the previous frame as $X_{dB}^{[-1]}(k)$. For every peak in the current compressed spectrum, normalized correlation is calculated with the previous compressed spectrum. The correlation operation takes into account all indexes (bins) of that peak, delimited by two consecutive minima. That is, if we denote the starting index of a particular minimum as:

$$i_{st} = i_{min}(x) \quad (E-37)$$

and the final index of that minimum as:

$$i_{end} = i_{min}(x+1) \quad (E-38)$$

we can express the correlation for that peak by:

$$M_{cor}(k) = \frac{\left(\sum_{j=i_{st}}^{i_{end}} X_{dB}(j) X_{dB}^{[-1]}(j) \right)^2}{\sum_{j=i_{st}}^{i_{end}} (X_{dB}(j))^2 \sum_{j=i_{st}}^{i_{end}} (X_{dB}^{[-1]}(j))^2} \quad k = i_{st}, \dots, i_{end} - 1 \quad (E-39)$$

The above three equations are then repeated for all minima, i.e., for $x=0, \dots, N_{min}-2$. The leading and the trailing bins of M_{cor} are set to zero. That is:

$$\begin{aligned} M_{cor}(k) &= 0 & k &= 0, \dots, i_{min}(0) - 1 \\ M_{cor}(k) &= 0 & k &= i_{min}(N_{min} - 1), \dots, 146 \end{aligned} \quad (E-40)$$

And the central part of the correlation is halved as:

$$M_{cor}(k) = 0.5 \cdot M_{cor}(k) \quad k = 45, \dots, 56 \quad (E-41)$$

to compensate the weights of compressed and uncompressed spectra for calculating the summation of the correlation map by equation E-43.

Let us call the array M_{cor} as "the correlation map". The correlation map has the same number of bins as X_{dB} . A graphical illustration of the correlation map is given in Figure E.2.

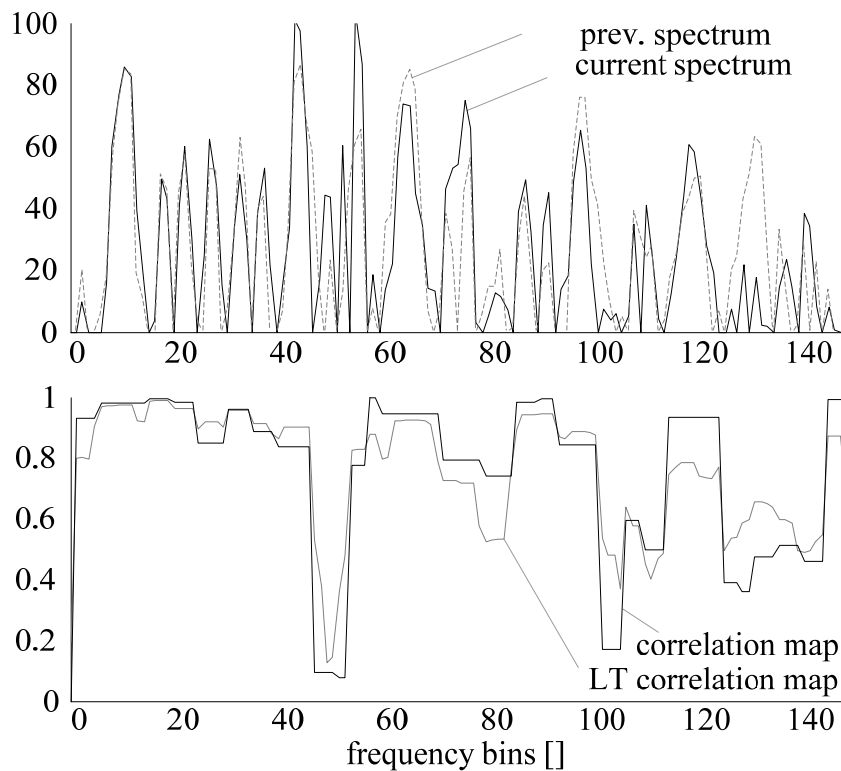


Figure E.2 – Calculation of the correlation map and the long-term correlation map

The correlation map of the current frame is used to update its long-term value, which is expressed by:

$$\overline{M}_{cor}(k) = \alpha_{map} \overline{M}_{cor}(k) + (1 - \alpha_{map}) M_{cor}(k) \quad k = 0, \dots, 146 \quad (E-42)$$

where $\alpha_{map} = 0.5$. The long-term correlation map, $\overline{M}_{cor}(k)$, is initialized to zero for all k . For an illustration of the long-term correlation map, see Figure E.2.

Finally, all bins of $\overline{M}_{cor}(k)$ are summed together by:

$$m_{sum} = \sum_{j=0}^{146} \overline{M}_{cor}(j) \quad (\text{E-43})$$

The decision about tonality is taken by subjecting m_{sum} to an adaptive threshold th_{tonal} . This threshold th_{tonal} is initialized to 52.5 and updated in every frame by:

$$\text{if } (m_{sum} > 52.5)$$

$$th_{tonal} = th_{tonal} - 0.5$$

else

$$th_{tonal} = th_{tonal} + 0.5$$

and is upper limited by 57.5 and lower limited by 47.5. Thus, it decreases when the correlation is relatively good indicating an active (tonal) segment and increases otherwise. When the threshold is lower, more frames will be classified as tonal, especially at the end of active periods. Therefore, the adaptive threshold may be viewed as a hangover.

The output of the tonality estimation is the information of whether the current frame is considered tonal or not. If m_{sum} is greater than th_{tonal} , the current frame is considered tonal ($tonal = 1$), otherwise it is non-tonal ($tonal = 0$). The tonality information is included also in the bitstream submitted to the decoder. The tonality decision is used further in the codec through a binary variable $tonal$. If the current frame is classified as tonal, the sinusoidal mode is used; if not, the generic mode is used.

E.6.5 Generic mode coding

The generic mode is used when the input frame is not considered to be tonal ($tonal = 0$). It utilizes the coded MDCT domain representation $\hat{M}_{16}(k)$ of the ITU-T G.729.1 wideband codec to code the high frequencies. The high frequency band (7-14 kHz) is divided into four sub-bands, and for every sub-band the most similar match with the selected similarity criteria is searched from the coded and envelope normalized wideband content. The most similar match is scaled with two scaling factors, the first one in linear domain and the second one in logarithmic domain, to obtain the synthesized high frequency content. This content is further improved with additional sinusoids, both within the first SWB extension layer, Layer 6mo, and within additional sinusoidal layers.

E.6.5.1 Obtaining MDCT domain synthesized WB content

The SWB extension utilizes the quantized 32 kbit/s MDCT domain representation $\hat{M}_{16}(k)$, $k = 0, \dots, 279$, of the ITU-T G.729.1 core codec.

E.6.5.2 Envelope normalization

The MDCT domain envelope of the low frequency content often evolves rapidly, which makes it difficult to find good matches for the high frequency content. In addition, there may be large dynamics differences, which make the quantization of the scaling factors more difficult. To avoid these problems, the envelope of the quantized WB content $\hat{M}_{16}(k)$ is normalized before further processing. The normalization is performed in logarithmic domain.

Firstly, the quantized WB content $\hat{M}_{16}(k)$ is divided into 35 sub-bands. Each sub-band $SB_k(i)$ consists of eight MDCT coefficients:

$$SB_k(i) = \hat{M}_{16}(8k + i) \quad i = 0, \dots, 7; \quad k = 0, \dots, 34 \quad (\text{E-44})$$

The arithmetic means of the first and second half of each sub-band $M_a^1(k)$ and $M_a^2(k)$ are calculated as follows:

$$M_a^1(k) = \frac{\sum_{i=0}^3 |SB_k(i)|}{4} \quad k = 0, \dots, 34 \quad (\text{E-45})$$

$$M_a^2(k) = \frac{\sum_{i=4}^7 |SB_k(i)|}{4} \quad k = 0, \dots, 34$$

Next, the geometric mean of $M_a^1(k)$ and $M_a^2(k)$ in logarithmic domain of each sub-band, $L(k)$ is obtained as:

$$L(k) = 20 \cdot \log_{10}(M_a^1(k) \cdot M_a^2(k)) \cdot 0.5 \quad k = 0, \dots, 34 \quad (\text{E-46})$$

Here, the square root calculation in linear domain is performed by multiplying by 0.5.

In the case where $M_a^1(k) \cdot M_a^2(k)$ is equal to zero and $(M_a^1(k))^2 + (M_a^2(k))^2$ is greater than zero, $L(k)$ is given by:

$$L(k) = 20 \cdot \log_{10}((M_a^1(k))^2 + (M_a^2(k))^2) \cdot 0.5 \quad k = 0, \dots, 34 \quad (\text{E-47})$$

for avoiding inputting zero-value to the logarithmic function. If $(M_a^1(k))^2 + (M_a^2(k))^2$ is equal to zero, then $L(k)$ is calculated by a logarithm of 0x0001 in the fixed point implementation.

The 35 geometric means, $L(k)$, $k = 0, \dots, 34$, are smoothed using moving average method with a span of seven samples. The envelope smoothed values $\tilde{L}(k)$ are obtained as:

$$\tilde{L}(k) = \begin{cases} L(k), & k = 0 \\ \frac{L(k-1) + L(k) + L(k+1)}{3}, & k = 1 \\ \frac{L(k-2) + \dots + L(k+2)}{5}, & k = 2 \\ \frac{L(k-3) + \dots + L(k+3)}{7}, & k = 3, \dots, 31 \\ \frac{L(k-2) + \dots + L(k+2)}{5}, & k = 32 \\ \frac{L(k-1) + L(k) + L(k+1)}{3}, & k = 33 \\ L(k), & k = 34 \end{cases} \quad (\text{E-48})$$

The smoothed envelope $\tilde{L}(k)$ is transformed from logarithmic domain to linear domain as follows:

$$\tilde{L}'(k) = 10^{(-1.0) \cdot 0.05 \cdot (\tilde{L}(k))} \quad k = 0, \dots, 34 \quad (\text{E-49})$$

The quantized WB coefficients $\hat{M}_{16}(k)$ are normalized by multiplying them by the smoothed envelope $\tilde{L}'(k)$ in order to obtain the normalized WB coefficients $\tilde{M}_{16}(k)$:

$$\tilde{M}_{16}(k) = \hat{M}_{16}(k) \cdot \tilde{L}'(k') \quad k = 8 \cdot k', \dots, 8 \cdot k' + 7 \quad k' = 0, \dots, 34 \quad (\text{E-50})$$

E.6.5.3 Sub-band search

The high-frequency (HF, 7-14 kHz) band of $M_{32}(k)$ is divided into four sub-bands as follows:

$$\begin{aligned} M_{32}^0(k) &= M_{32}(k + 280), \quad k = 0, \dots, 39 \\ M_{32}^1(k) &= M_{32}(k + 320), \quad k = 0, \dots, 69 \\ M_{32}^2(k) &= M_{32}(k + 390), \quad k = 0, \dots, 69 \\ M_{32}^3(k) &= M_{32}(k + 460), \quad k = 0, \dots, 99 \end{aligned} \quad (\text{E-51})$$

The widths of the sub-bands, d^j , are 40, 70, 70 and 100 MDCT domain samples for $j = 0, \dots, 3$, respectively. For every sub-band $M_{32}^j(k)$ there is a search band $k = k^j, \dots, k^j + w^j - 1$ which defines from where the best match is searched, i.e., for band j there are w^j different positions.

The start positions for the search in even sub-bands are fixed while the start positions in odd sub-bands vary adaptively depending on the best match position in the lower even sub-band.

The start position k^j and the search range w^j of each sub-band are set as follows:

$$k^j = \begin{cases} 0 & j = 0 \\ \text{BestIdx}^{j-1} + d^{j-1} - \frac{w^j}{2} & j = 1 \\ 0 & j = 2 \\ \text{BestIdx}^{j-1} + d^{j-1} - \frac{w^j}{2} & j = 3 \end{cases} \quad (\text{E-52})$$

$$w^j = \begin{cases} 240 & j = 0 \\ 128 & j = 1 \\ 210 & j = 2 \\ 128 & j = 3 \end{cases} \quad (\text{E-53})$$

where BestIdx^j is the best match position for band j determined by the following procedure. Here, if k^j becomes a negative value in equation E-52, k^j is set to 0. And if k^j becomes larger than $280 - d^j - w^j$, k^j is set to $280 - d^j - w^j$.

The best match is searched for every sub-band j as follows. The correlation for index k' is computed as:

$$\text{corr}(k') = \sum_{k=0}^{k=d^j-1} M_{32}^j(k) \tilde{M}_{16}(k^j + k' + k) \quad k' = 0, \dots, w^j - 1 \quad (\text{E-54})$$

Similarly, the energy for index k' is obtained as

$$\text{Ene}(k') = \sum_{k=0}^{k=d^j-1} \tilde{M}_{16}(k^j + k' + k)^2 \quad k' = 0, \dots, w^j - 1 \quad (\text{E-55})$$

The actual similarity measure used is of the form

$$S(k') = \left| \frac{\text{corr}(k')}{\sqrt{\text{Ene}(k')}} \right| \quad k' = 0, \dots, w^j - 1 \quad (\text{E-56})$$

The task is to find k' which maximizes $S(k')$. The complexity of the implementation can be reduced by squaring $S(k')$, which removes the absolute value signs as well as the square root from the denominator. The best match is now searched efficiently for sub-band j using:

```

BestIdxj = 0
lagCorr = 0
lagEnergy = 1e30
for k' = 0 to wj - 1
    if (Ene(k') > 0)
        if (lagCorr2 Ene(k') < corr(k')2 lagEnergy)
            BestIdxj = k'
            lagCorr = corr(k')
            lagEnergy = Ene(k')
        end
    end
end
end

```

The best match index BestIdx^j is packed into the bitstream as the parameter LagIndex^j .

E.6.5.4 First scaling

Once the best match has been found in the sub-band search, two scaling factors are used to achieve the perceptually most similar match with the original sub-band $M_{32}^j(k)$. The first scaling factor $\alpha_1(j)$ operates on the linear domain to match the high amplitude peaks in the spectrum. It is obtained for sub-band j as:

$$\alpha_1(j) = \frac{\text{corr}(\text{BestIdx}^j)}{\text{Ene}(\text{BestIdx}^j)} \quad j = 0, \dots, 3 \quad (\text{E-57})$$

where BestIdx^j is the found index with best match for sub-band j . Scaling factor $\alpha_1(j)$ can get both positive and negative signs.

The signs and the absolute values of the scaling factors are quantized separately. The sign values are defined as:

$$\text{Sign_subband}(j) = \begin{cases} 0, & \alpha_1(j) \geq 0 \\ 1, & \text{otherwise} \end{cases} \quad (\text{E-58})$$

The absolute values are quantized in the logarithmic domain. The transform to logarithmic domain is performed as follows:

$$\alpha_1^{\text{Log}}(j) = \begin{cases} 0, & \alpha_1(j) = 0 \\ \log_{10}(|\alpha_1(j) + 1e-15|), & \text{otherwise} \end{cases} \quad (\text{E-59})$$

The four logarithmic domain scaling factors are now grouped into two vectors:

$$\begin{aligned} V_1 &= [\alpha_1^{\text{Log}}(0) \quad \alpha_1^{\text{Log}}(2)] \\ V_2 &= [\alpha_1^{\text{Log}}(1) \quad \alpha_1^{\text{Log}}(3)] \end{aligned} \quad (\text{E-60})$$

E.6.5.5 Vector quantization with an ordered first component codebook

The two vectors are quantized using 2-dimensional vector quantization (VQ) using the 8 bits codebook C_2_8. The nearest neighbour search procedure is common to the quantization of the logarithmic domain scaling factors, second scaling factors and sinusoids discussed in the following subclauses. The generic search procedure is described here.

Given the input vector $\mathbf{v}$, a binary search is performed in the codebook that is increasingly ordered after the first component. The binary search procedure returns the lowest index $idx1$ from the codebook corresponding to the codevector whose first component is greater than the first component of the input vector. Starting from the codevector whose index is given by the binary search, a modified partial distortion search approach is performed such that the search is stopped altogether if the partial distortion calculated only for the first component is larger than the current minimum distortion. The direction of search in the ordered codebook is upwards from index $idx1$ and downwards from $idx1-1$. If $idx1$ is equal to the number of codevectors from the codebook minus 1, only the downward search is performed. Conversely, if $idx1$ is equal to 0, only the upward search is performed. When both directions are tested, the winner is selected between the resulting codevectors from the two modified partial search distortions.

As it will be mentioned in the next clauses, this procedure is also applied for 1-, 3-, and 4-dimensional codebooks. Yet, for the 1-dimensional quantizer using the 4-bit codebook E_1_4, after the binary search, only the distortions for codebook indices $idx1-1$ and $idx1$ need to be computed to select the best index. It should also be noted that for the 3-dimensional VQ and for one of the 4-dimensional VQ using the 8-bit quantizer codebooks E_3_8 and E_4_8, respectively, the generic procedure is performed twice, first to select the best codevector among the first 128 codevectors, and then to select the best codevector among the last 128 codevectors. The winner is then selected between the resulting codevectors from the two generic searches (see clause E.6.6.2).

As a result of quantization we have quantized log domain parameters $\hat{\alpha}_1^{Log}(j)$. These are again transformed to the linear domain using:

$$\hat{\alpha}_1(j) = 10^{\hat{\alpha}_1^{Log}(j)} \quad (\text{E-61})$$

In addition, the original signs are returned using:

$$\hat{\alpha}_1(j) = \begin{cases} -\hat{\alpha}_1(j), & \text{if } (\text{Sign_subband}(j) \neq 0) \\ \hat{\alpha}_1(j), & \text{otherwise} \end{cases} \quad (\text{E-62})$$

E.6.5.6 Second scaling

The second scaling factor operates on the logarithmic domain and is used to provide a better match with the energy and the logarithmic domain shape. For computing the second scaling factor, logarithmic domain representations are needed both for the original high frequency band $M_{32}(k)$ as well as for the so-far synthesized high frequency content.

The so-far synthesized high frequency content is obtained for every sub-band $j = 0, \dots, 3$ as:

$$\dot{M}_{32}^j(k) = \tilde{M}_{16}(k^j + k) \hat{\alpha}_1(j) \quad k = 0, \dots, d^j - 1 \quad (\text{E-63})$$

where k^j is generated from the parameter $LagIndex^j$ which is determined by sub-band search (in clause E.6.5.3) as follows:

$$k^j = \begin{cases} LagIndex^j & j = 0, 2 \\ \min(\max(0, LagIndex^{j-1} + d^{j-1} - \frac{w^j}{2}), 280 - d^j - w^j) + LagIndex^j & j = 1, 3 \end{cases} \quad (E-64)$$

The overall so-far synthesized high band content consists of the four sub-bands as follows:

$$\dot{M}_{32}(k) = \begin{cases} \dot{M}_{32}^0(k-280), & k = 280, \dots, 319 \\ \dot{M}_{32}^1(k-320), & k = 320, \dots, 389 \\ \dot{M}_{32}^2(k-390), & k = 390, \dots, 459 \\ \dot{M}_{32}^3(k-460), & k = 460, \dots, 559 \end{cases} \quad (E-65)$$

For the even number k , the original high frequency band of $M_{32}(k)$ is transformed to logarithmic domain as:

$$\hat{M}_{32}(k) = 20 \cdot \log_{10}(|M_{32}(k) + 1e-15|) \quad k = 280, 282, \dots, 558 \quad (E-66)$$

The overall $\hat{M}_{32}(k)$ consist of the four sub-bands as follows:

$$\begin{aligned} \hat{M}_{32}^0(k) &= \hat{M}_{32}(k+280), & k &= 0, 2, \dots, 38 \\ \hat{M}_{32}^1(k) &= \hat{M}_{32}(k+320), & k &= 0, 2, \dots, 68 \\ \hat{M}_{32}^2(k) &= \hat{M}_{32}(k+390), & k &= 0, 2, \dots, 68 \\ \hat{M}_{32}^3(k) &= \hat{M}_{32}(k+460), & k &= 0, 2, \dots, 98 \end{aligned} \quad (E-67)$$

For the even number k , the synthesized high band is also transformed to the logarithmic domain, i.e.,

$$\hat{\dot{M}}_{32}(k) = 20 \cdot \log_{10}(|\dot{M}_{32}(k) + 1e-15|) \quad k = 280, 282, \dots, 558 \quad (E-68)$$

which equals to:

$$\hat{\dot{M}}_{32}^j(k) = 20 \cdot \log_{10}(|\dot{M}_{32}^j(k) + 1e-15|) \quad j = 0, \dots, 3 \quad k = 0, 2, \dots, d^j - 2 \quad (E-69)$$

For every logarithmic domain sub-band, the maximum value is searched:

$$m^j = \max\left(-10000, \max_k(\hat{\dot{M}}_{32}^j(k))\right) \quad j = 0, \dots, 3 \quad k = 0, 2, \dots, d^j - 2 \quad (E-70)$$

The position satisfying equation E-70 for each sub-band is denoted as $k_{\max}^j$. Correlation and energy are computed for logarithmic domain sub-bands from which the maximum value has been subtracted. The operations are performed for the even samples of the logarithmic so-far synthesized high band content:

$$LogCorr = \sum_k (\hat{\dot{M}}_{32}^j(k) - m^j)(\hat{\dot{M}}_{32}^j(k) - m^j) \quad j = 0, \dots, 3 \quad k = 0, 2, \dots, d^j - 2 \quad (E-71)$$

$$LogEne = \sum_k (\hat{\dot{M}}_{32}^j(k) - m^j)^2 \quad j = 0, \dots, 3 \quad k = 0, 2, \dots, d^j - 2 \quad (E-72)$$

Now if $\text{LogEne} > 0$, the second scaling factor is obtained as follows:

$$\alpha_2(j) = \frac{\text{LogCorr}}{\text{LogEne}} \quad (\text{E-73})$$

If $\text{LogEne} = 0$, $\alpha_2(j)$ is set to 1. In addition $\alpha_2(j)$ is forced to be positive

if $\alpha_2(j) < 0$

$$\alpha_2(j) = 0$$

end

The second scaling factors are quantized using vector quantization. First, a vector is created as:

$$V = [\alpha_2(0) \alpha_2(1) \alpha_2(2) \alpha_2(3)] \quad (\text{E-74})$$

The 4-dimensional vector V is quantized using an 8-bit codebook (D_4_8) that is increasingly ordered after the first component. The nearest neighbour search is performed similarly to the procedure described in clause E.6.5.5.

As a result of quantization, we have quantized parameters $\hat{\alpha}_2(j)$. The high frequency content synthesized with the two scaling factors, $\ddot{M}_{32}(k)$, can now be computed.

First, the selection flag $F_m(k)$ is defined by the following equation:

$$F_m(k) = \begin{cases} 1 & \text{if } ((P_m(j) - R_m(j) \leq k \leq P_m(j) + R_m(j)) \\ & \text{or } (k = 280, 282, \dots, 558)) \\ 0 & \text{otherwise} \end{cases} \quad j = 0, \dots, 3 \quad (\text{E-75})$$

where $P_m(j)$ is defined as:

$$P_m(j) = \begin{cases} 280 + k_{\max}^j & j = 0 \\ 280 + \sum_{i=0}^{j-1} d^i + k_{\max}^j & j = 1, \dots, 3 \end{cases} \quad (\text{E-76})$$

$R_m(j)$ is a predefined array of constants:

$$R_m(j) = \begin{cases} 5 & j = 0 \\ 15 & j = 1, 2 \\ 21 & j = 3 \end{cases} \quad (\text{E-77})$$

Here, the selection flag $F_m^j(k)$ for each sub-band is defined as:

$$\begin{aligned} F_m^0(k) &= F_m(k + 280), & k &= 0, \dots, 39 \\ F_m^1(k) &= F_m(k + 320), & k &= 0, \dots, 69 \\ F_m^2(k) &= F_m(k + 390), & k &= 0, \dots, 69 \\ F_m^3(k) &= F_m(k + 460), & k &= 0, \dots, 99 \end{aligned} \quad (\text{E-78})$$

Next, the odd samples of the so-far synthesized high band content are transformed to the logarithmic domain, when the selection flag $F_m^j(k)$ is equal to 1 as follows: