# 6 Functional description of the Decoder

## 6.1 LP-based Decoding

### 6.1.1 General LP-based decoding

The LSF parameters are decoded from the received bitstream and converted to LSP coefficients and subsequently to LP coefficients. The interpolation principle, described in subclause 5.1.9.6, is used to obtain interpolated LSP vectors for all subframes, i.e. 4 subframes in case of 12.8 kHz internal sampling rate and 5 subframes in case of 16 kHz sampling rate. Then, the excitation signal is reconstructed and post-processed before performing LP synthesis (filtering with the LP synthesis filter) to obtain the reconstructed signal. The reconstructed signal is then de-emphasized (an inverse of the pre-emphasis applied at the encoder). Finally, a post-processing is applied for enhancing the format and harmonic structure of signal as well as the periodicity in the low frequency region of the signal. The signal is then up-sampled to the output sample rate. Finally, the high-band signal is generated and added to the up-sampled synthesized signal to obtain a full-band reconstructed signal (output signal).

#### 6.1.1.1 LSF decoding

##### 6.1.1.1.1 General LSF decoding

Depending on the predictor allocation per mode, like specified at encoder side in subclause 5.2.2.1.3 one first bit is read to select between safety net or predictive mode for the switched safety net/predictive cases. The bit value of one corresponds to safety net and value zero corresponds to predictive mode. The following bits are read in groups of a number equal to the stage sizes corresponding to each coding mode as specified in subclause 5.2.2.1.4 and the codevectors are retrieved from the corresponding codebooks. The last bits correspond to the lattice codevector, having the index . The LSF residual after the first non-structured, optimized VQ was quantized by splitting the vector into two subvectors. The index  was obtained as a combined index of the two indexes corresponding to the first and the second subvector. The two indexes are retrieved as follows:

 (1409)

 ()

These indexes correspond each to a scale index, leader class index and leader vector permutation index. For each of the two indexes corresponding to the 8-dimensional subvectors the following operations are applied. The scale offset is determined by finding out the largest scale offset that is smallest than the index . The corresponding scale offset is removed from each of . Similarly the leader offset is calculated and removed for each of the two indexes. The index of the scale offset gives the index of the scale, , and the index of the leader offset gives the index of the leader class, . The remaining index values are . The sign index,  and the leader index  are obtained

 ()

 ()

where is the cardinality of unsigned permutations for the leader class , given in subclause 5.2.2.1.4. The indexes  and  are decoded using the position decoding based on counting the binomial coefficients and the sign decoding described in [26].

Decoding of the index corresponding to the unsigned permutation of the leader vector goes as follows. Knowing the leader class index, the number of distinct non zero values and the amount of each of these values which are tabulated (see subclause 5.2.2.1.4) can be determined. The used leader classes defined in subclause 5.2.2.1.4 have at most 4 distinct values. If there is a single value, , in the leader class corresponding to the decoded leader class index, all decoded vector components have the same value

 ()

where is the subvector dimension.

If there are two distinct values *,* in the decoded leader vector, each appearing  and  times respectively, the decoded vector is initialized with

. ()

The leader vector permutation index is interpreted using binomial coefficients decoding. The positions of the  values  are determined within a vector of length . The position of the first , is determined such that

. ()

If then . The position of the second  value, , is determined similarly for an updated index

 ()

an updated number vector length,  instead of , and an updated number of values,  instead of .

The procedure follows until the positions of all *v*0 values are determined. Once these positions are known the values  are inserted in the vector  at the corresponding positions.

If there are 3 distinct values  having  number of occurrences respectively, the decoded vector is initialized with:

. ()

Out of , two subindexes are obtained:

. ()

. ()

The positions of the values  are determined by binomial decoding of the index  considering  positions out of and the values  are inserted in the vector . The decoding is performed according to equations (1208) and (1209). The positions for values *v*0 are obtained by binomial decoding of the index *Li1*, considering *k*0 positions out of *S.*

If there are 4 distinct values the vector is initialized with

. ()

The index  is divided into:

. ()

. ()

. ()

. ()

The positions for values  are obtained by binomial decoding the index  for  position out of . The positions for values  are obtained by binomial decoding of the index  for  positions out of . The positions for values  are obtained by binomial decoding of the index  for  positions out of .

The obtained subvectors are multiplied with the corresponding scales and component wise multiplied with the off-line computed standard deviations. The standard deviations are individually estimated for each coding mode and bandwidth. The result corresponds to the codevector from the last stage of the LSF quantizer. The codevectors from all stages are added together.

If the coding mode corresponds to a safety net only mode, or if it corresponds to a switched safety net/AR predictive mode and the safety net mode has been selected at the encoding stage, a vector representing the component wise mean for the current coding mode is added to the sum of codevectors and the result represents the decoded LSF vector. The decoded LSF vector is thus given by:

, for =0,…, -1 (1425)

where is the LSF vector for current frame , *lk*(*i*), *i*=0, *M*-1 is the codevector obtained at stage  out of the  quantization stages and  is the mean LSF vector for the current coding mode.

If AR predictive mode was selected at the encoding stage, the decoded LSF vector is given by:

, for =0,…, -1. (1426)

If MA predictive mode was selected at the encoding stage, based on the coding mode, the decoded LSF vector is given by:

, for =0,…, -1. (1427)

where  is the quantization error at the previous frame .

##### 6.1.1.1.2 LSF decoding for voiced coding mode at 16 kHz internal sampling frequency

The VC mode at the 16 kHz internal sampling frequency has two decoding rates: 31 bits per frame and 40 bits per frame. The VC mode is decoded by a 16-state and 8-stage BC-TCVQ. figure 88 shows the decoder of the predictive BC-TCVQ with safety-net using an encoding rate of 31 bits. The 31bit LSF decoding performed by the predictive BC-TCVQ with safety-net proceeds as follows. First, one bit is decoded at the *Scheme selection* block. This bit defines whether the predictive scheme or the safety-net scheme is used.

For the safety-net scheme,  is decoded by equation (1428),

, for =2,…, /2 (1428)

where the prediction residual, , is decoded by the 1st BC-TCVQ.

If the predictive scheme is used, the prediction vector  is obtained using (1429):

, for =0,…, -1 (1429)

where are the selected AR prediction coefficients for the VC mode at 16kHz isf, M is the LPC order, and .

The decoding of  is performed as given by equation (1430),

, for =2,…, /2 (1430)

where the prediction residual, , is decoded by the 2nd BC-TCVQ.

The quantized LSF vector  for the predictive scheme is calculated by equation (1431),

, for =0,…, -1 (1431)

where  is the mean vector for VC mode and 

The quantized LSF vector for the safety-net scheme is calculated by equation (1432).

, for =0,…, -1 (1432)



Figure 88: Block diagram of the decoder for the predictive BC-TCVQ with safety-net for an encoding rate of 31 bits per frame

Figure 89 shows the decoder of the predictive BC-TCVQ with safety-net for an encoding rate of 40 bits per frame. The 40-bit LSF decoding using the predictive BC-TCVQ with safety-net is performed as follows. The scheme selection and the decoding method of BC-TCVQ for both the predictive and safety-net schemes are the same as those of the 31-bit LSF decoding.  and  are decoded by the 3rd and 4th SVQ decoding respectively. The quantized LSF vector  for the predictive scheme is calculated according to equation (1433),

, for =0,…,-1 (1433)

where  is the output of the 2nd BC-TCVQ and the 2nd intra-frame prediction.

The quantized LSF vector for the safety-net scheme is calculated by equation (1434),

, for =0,…, -1 (1434)

where  is the output of the 1st BC-TCVQ and 1st intra-frame prediction.



Figure 89: Block diagram of the decoder for the predictive BC-TCVQ/SVQ with safety-net for an encoding rate of 40 bits per frame

#### 6.1.1.2 Reconstruction of the excitation

##### 6.1.1.2.1 Reconstruction of the excitation in GC and VC modes and high rate IC/UC modes

6.1.1.2.1.1 Decoding the adaptive codebook vector

The received adaptive codebook parameters (or pitch parameters) are the closed-loop pitch, , and the pitch gain, (adaptive codebook gain), transmitted for each subframe, serve to compute the adaptive codevector, .

6.1.1.2.1.2 Pulse index decoding of the 43-bit algebraic codebook

The joint indexing decoding procedure of three pulses on two tracks is described as follows:

In the decoder side, the de-indexing procedure is as below for pulses,  positions on the track:

1. 24 bits are extracted from the received bit-stream and then decoded as the temporary index . If  is smaller than *THR* which is the same as the encoder side, the joint index equals to . If  is bigger than or equal to *THR*, 1 more bit will be extracted from the bit-stream as *Bit*. Then the global index  is adjusted as: . Then the joint index  is computed by subtracting *THR* from :

 (1435)

1. Decompress the joint index into the two index for each track:

 (1436)

 (1437)

1. Decoding the index for each track(and ) as below:
2. determining the quantity of pulse positions according to the first index 

As the offset index  is saved in a table (available in encoder and decoder), and each offset index in the table indicates the unique number of pulse positions in the track. So  can be decoded from the index easily. Then the number of pulse position , the sign index and  are obtained.

1. As we know the number of pulse position  and index , the indexand  can be decoded based on permutation method from the index , and each pulse position is also decoded from and .Separating and obtaining the second index and the third index in the following way:

 (1438)

 (1439)

wherein represents the second index, represents the third index, represents the quantity of the positions with pulse on it, refers to taking the remainder, and “Int” refers to taking the integer

1. determining the distribution of the positions with a pulse on the track according to the second index;

the is obtained, the following calculation process is applied at the decoder:

(1) , ..., and  are subtracted from one by one.

 (1440)

until the remainder changes from a positive number to a negative number, where is the total quantity of positions on the track, is the quantity of positions with pulses, , and C refers to calculating the combination function. The , namely, the serial number of the first position with a pulse(s) on the position, is recorded, where.

(2) If , , ..., and  are further subtracted from one by one until the  remainder changes from a positive number to a negative number. The namely, the serial number of the second position with a pulse(s) on the position, is recorded, where .

(3) And so on, , ..., and  are further subtracted from  one by one until the remainder changes from a positive number to a negative number, where . The  namely, the serial number of the n+1 position with a pulse(s) on the position, is recorded, where .

(4) The decoding of the is completed, and is obtained.

1. determining the quantity of pulses in each position with pulses according to the third index;

For each track, according to the third index , determine the number of pulses on each position that has a pulse. the  is obtained, the following calculation process is applied at the decoder:

(1) is calculated from a smaller value to a greater value, where: , ,, and C refers to calculating the combination function. The last value that lets be greater than zero is recorded as the position of the first pulse on the track.

(2) If , is further calculated from a smaller value to a greater value, where ; and the last value that lets be greater than zero is recorded as the position of the second pulse on the track.

(3) By analogy, is calculated from a smaller  value to a greater value, where: , and ; and the last  value that lets be greater than zero is recorded as the position for the (h+1)*th* pulse(h+1 is an ordinal number) on the track.

(4) The decoding of the is completed, and  is obtained.

1. After obtain , mean on each position  have a pulse, if , mean on the position  have more pulses. The is the result after subtract value “1” from the number of pulses in each pulse position, so value “1” is need to be added back to  position, and is rebuilt as following







1. By now all the pulse positions, the quantity of pulses in each pulse position and associated signs are decoded, so the pulses on each track is reconstructed.

6.1.1.2.1.3 Mulit-track joint decoding of pulse indexing

All the muti-track joint decoding step is described as following:

1. extracting the , ,, and from the stream;
2. Get the parameter from the table 35 according to the pulse number of each track, include the index *bitst*, *Hi\_Bit\_bitst*, *Hi\_Bit\_ranget*, *re-back\_bitst*,
3. Extract and from , extract and from , extract and from , extract  and from .
4. From ,, ,  and , , ,  and are decoded out.
5. The  is combined with  and obtain *,*  is combined with  and obtain , then can be get as following:

 (1441)

 (1442)

1. The  is combined with  and obtain , then can be get as following:

 (1443)

 (1444)

1. The  is combined with  and obtain , then , can be get as following:

 (1445)

 (1446)

1. Combine , , ,with , , , , and get the index of each track.

6.1.1.2.1.4 Decoding the algebraic codebook vector

The received algebraic codebook index is used to extract the positions and amplitudes (signs) of the excitation pulses and to find the algebraic codevector . If the integer part of the pitch lag is less than the subframe size 64, the pitch sharpening procedure is applied, which translates into modifying by filtering it through the adaptive pre-filter  which further consists of two parts: a periodicity enhancement part , where is the integer part of the pitch lag representing the fine spectral structure of the speech signal, and a tilt part, where is related to the voicing of the previous subframe and is bounded by [0.28, 0.56] at 16.4 and 24.4 kbps, and by [0.0; 0.5] otherwise.

The periodicity enhancement part of the filter colours the spectrum by damping inter-harmonic frequencies, which are annoying to the human ear in case of voiced signals.

Depending on bitrates and coding mode, and the estimated level of background noise, the adaptive pre-filter also includes a filter based on the spectral envelope, which colours the spectrum by damping frequencies between the formant regions. The final form of the adaptive pre filter is given by

 (1447)

where and if Hz and and if Hz.

6.1.1.2.1.5 Decoding of the combined algebraic codebook

At 32 kbps and 64 kbps bit-rates, the pre-quantizer excitation contribution is obtained from the received pre-quantizer parameters as follows. The contribution from the pre-quantizer is obtained by first de-quantizing the decoded (quantized) spectral coefficients using an AVQ decoder and applying the iDCT to these de-quantized spectral coefficients. Further the pre-emphasis filter  is applied after the iDCT to form the pre-quantizer contribution . The pre-quantizer contribution  then scales using the quantized pre-quantizer gain  to form the pre-quantizer excitation contribution.

The same above procedure applies for decoding GC, TC and IC mode at 32 kbps and 64 kbps with the exception of non-harmonic signals at 32kbps GC mode where the iDCT stage is omitted. It is noted that at the decoder, the order of codebooks and corresponding codebook stages during the decoding process is not important as a particular codebook contribution does not depend on or affect other codebook contributions. Thus the codebook arrangement in the IC mode is identical to the GC mode codebook arrangement. The pre-quantizer gain in GC and TC mode is obtained by

 ()

where  is the decoded normalized pre-quantizer gain and  predicted algebraic codevector energy.

In IC mode, the de-quantizer gain is obtained by

 ()

where  is the quantized algebraic codebook gain.

6.1.1.2.1.6 AVQ decoding

The reading of the AVQ parameters from the bitstream is complementary to the insertion described in subclause 5.2.3.1.6.9.3. The codebook numbers  are used to estimate the actual bit-budget needed to encode AVQ parameters at the decoder and the number of unused AVQ bits is computed as a difference between the allocated and actual bit budgets.

6.1.1.2.1.6.1 Decoding of AVQ parameters

The parameters decoding involves decoding the AVQ parameters describing each 8-dimensional quantized sub‑bands  of the quantized spectrum . The  comprise several sub-bands (8 in case of combined algebraic codebook), each of 8 samples. The decoded AVQ parameters for each sub‑band  comprise:

* the codebook number ,
* the vector index  ,
* and, if the codevector (i.e. lattice point) is not in a base codebook, the Voronoi index .

The unary code for the codebook number , is first read from the bitstream and  is determined. From the codebook number , the base codebook and the Voronoi extension order are then obtained. If , there is no Voronoi extension () and the base codebook is . If  the base codebook is either *Q*3 ( even) or *Q*4 ( odd) and the Voronoi order (1 or 2) is also determined ( if ; , otherwise).

Then, if , the vector index , coded on  bits is read from the bitstream and the base codevector  is decoded.

After the decoding of the base codevector, if the Voronoi order  is greater than 0, the Voronoi extension index  is decoded to obtain the Voronoi extension vector . The number of bits in each component of index vector  is given by the Voronoi extension order , and the scaling factor  of the Voronoi extension is given by .

Finally, from the scaling factor , the Voronoi extension vector  and the base codebook vector , each 8-dimensional AVQ sub-band  is computed as:

 ()

In case of decoding the pre-quantizer, resp. de-quantizer, contribution from subclause 6.1.1.2.1.3, the decoded sub-band blocks of corresponds to the decoded spectrum coefficients , resp. .

6.1.1.2.1.6.2 De-indexing of codevector in base codebook

The index decoding of the codevector  is done in several steps. First, the absolute leader and its offset are identified by comparing the index with the offset in the look‑up table. The offset is subtracted from the index to produce a new index. From this index, the sign index and the absolute vector index are extracted. The sign index is decoded and the sign vector is obtained. The absolute vector index is decoded by using a multi-level permutation-based index decoding method and the absolute vector is obtained. Finally, the decoded vector is reconstructed by combining the sign vector with the absolute vector.

6.1.1.2.1.6.2.1 Sign decoding

The sign vector is obtained by extracting from left to right all the sign bits for non-zero elements in the absolute vector. The bit number of the sign code is read from the (). If the bit number of the sign index is not equal to the number of the non-zero elements in the decoded absolute vector, the sign of the last non-zero element is recovered.

6.1.1.2.1.6.2.2 Decoding of the absolute vector and of its position vector

The decoding method of the absolute vector index is described as follows:

1. The absolute vector index is decomposed into several mid-indices for each level from lowest level to highest level. The absolute vector index is the starting value for the lowest level. The mid-index of each lower level is obtained by dividing the absolute vector index by the possible index value count, , the quotient is the absolute vector index for the next lower level. The remainder is the middle index, , for the current level.
2. The  of each lower level is decoded based on a permutation and combination function and the position vector of each lower level vector related to its upper level vector is obtained.

Finally, one-by-one from the lowest level to the highest level, each lower level absolute vector is used to partly replace the upper level absolute vector elements according to the position parameter. The highest level vector is the decoded output absolute vector. A example of the  absolute vector partly replace the absolute vector elements is give as following:



Figure 90: Replacing example between and  for .

6.1.1.2.1.6.2.3 Position vector decoding

To obtain the position vector from the middle index in each lower level, the algorithm uses a permutation and combination procedure to estimate the position sequence. The procedure is as follows:

1) Increment the  value beginning from zero, until  is not more than .

2) Let  be the first position, and subtract from the .

3) Increase , beginning from , until  is not more than , where  is the position decoded at the previous step.

4) Let  be the position number , and subtract from the .

5) Repeat steps 3 and 4 until all positions are decoded for the current level position sequence.

6.1.1.2.1.6.2.4 Absolute vector decoding

For the lowest level, the absolute vector only includes one type of element whose value can be obtained from the decomposition order column in the table of subclause 5.2.3.1.6.9.3.2. The lowest level absolute vector is passed to the next level and at the next step another type of element is added. This new element is obtained from the decomposition order column in the table of subclause 5.2.3.1.6.9.3.2. This procedure is repeated until the highest level is reached.

6.1.1.2.1.6.2.5 Construction of the output codevector in base codebook

Constructing the 8-dimensional output codevector in the base codebook is the final step of the decoding procedure. The codevector  is obtained by combining the sign vector with the absolute vector. If the bit number of the sign index is not equal to the number of the non-zero elements in the decoded absolute vector, the sign of the last non-zero element is recovered. The recovery rule, based on the *RE*8 lattice property, is as follows: if the sum of all output vector elements is not an integer multiple of 4, the sign of the last element is set to negative.

6.1.1.2.1.7 Decoding the gains

6.1.1.2.1.7.1 Decoding memory-less coded gains

Before calculating the adaptive and algebraic codebook gain in each subframe, the predicted algebraic codevector energy, , is decoded for the whole frame.

Now, let denote the algebraic codebook excitation energy in dB in a given subframe, which is given by

 ()

In the equation above, is the pre-filtered algebraic codevector.

A predicted algebraic codebook gain is then calculated as

 ()

An index is then retrieved from the bitstream representing a jointly-quantized adaptive codebook gain along with a correction factor. The quantized adaptive codebook gain, , is retrieved directly from the codebook and the quantized algebraic codebook gain is given by

 (1453)

whereis the decoded correction factor.

Note that no prediction based on parameters from past frames is used. This increases the robustness of the codec to frame erasures.

6.1.1.2.1.7.2 Decoding memory-less joint coded gains at lowest bit-rates

For the lowest bitrates of 7.2 and 8.0 kbps, slightly different memory-less joint gain coding scheme is used.

Similarly as in the encoder, the estimated (predicted) gain of the algebraic codebook in the first subframe is given by

 ()

where *CT* is the coding mode, selected for the current frame in the pre-processing part, and  is the energy of the filtered algebraic codevector. The inner term inside the logarithm corresponds to the gain of innovation vector. The only parameter in the equation above is the coding mode *CT* which is constant for all subframes of the current frame. The superscript [0] denotes the first subframe of the current frame.

In all subframes following the first subframe the estimated value of the algebraic codebook gain is given by

 ()

where *k*=1,2,3. Note, that the terms in the first and in the second sum of the exponent, there are quantized gains of algebraic and adaptive excitation of previous subframes, respectively. Note that the term including the gain of innovation vector  is not subtracted. The reason is in the use of the quantized values of past algebraic codebook gains which are already close enough to the optimal gain and thus it is not necessary to subtract this gain again.

The gain de-quantization in the decoder is done by retrieving the codevector [;] according to the index receing in the bitstream. The quantized value of the fixed codebook gain is then calculated as

 ()

6.1.1.2.1.7.3 Decoding scalar coded gains at highest bit-rates

As described in subclause 6.1.1.2.1.3.1, before calculating the adaptive and algebraic codebook gain in each subframe, the predicted algebraic codevector energy, , is decoded for the whole frame. Then two indexes are retrieved from the bitstream and used to decode the adaptive codebook gain and a correction factor. The decoded algebraic codebook gain is further obtained using equation (1453).

6.1.1.2.1.8 Reconstructed excitation

The total excitation in each subframe is constructed by

 ()

where is the pre-filtered algebraic codevector.

In case that combined algebraic codebook is used, the total excitation is each subframe is constructed by

 ()

The excitation signal, , is used to update the contents of the adaptive codebook for the next frame. The excitation signal, , is then post processed as described in subclause 7.1.2.4 to obtain the post-processed excitation signal , which is finally used as an input to the synthesis filter.

##### 6.1.1.2.2 Reconstruction of the excitation in TC mode

In TC mode, the TC frame configuration (subclause 6.8.4.2.2) is decoded first. Then, the adaptive excitation signal is either a zero vector, a glottal-shape codevector or an adaptive codebook vector. In a subframe where the glottal-shape codebook is used, the reconstruction of the glottal-shape codevector is done using the received TC parameters as described in subclause 6.8.4.2.1. In a subframe where the adaptive codebook is used, the adaptive codevector is found as described in subclause 7.1.2.1.1. In all subframes after the one where the glottal-shape codebook is used, a low-pass filtering is applied and the filtered adaptive excitation is found as .

If a subframe contains a zero adaptive excitation vector, only the algebraic codebook gain is decoded using a 2-bit or 3-bit scalar quantizer (described in subclause 6.8.4.2.4). Otherwise, the adaptive and algebraic codebook gains are decoded as in GC and VC modes (described in subclause 7.1.2.1.3).

Finally, the reconstructed excitation is computed as described in subclause 7.1.2.1.4.

##### 6.1.1.2.3 Reconstruction of the excitation in UC mode at low rates

6.1.1.2.3.1 Decoding the innovative vector

In UC mode, the signs and indices of the two random vectors are decoded and the excitation is reconstructed as in subclause 5.2.3.3.1. The correction of the random codebook tilt is used as described in subclause 5.2.3.3.2.

6.1.1.2.3.2 Decoding the random codebook gain

In UC mode, only the random codebook gain is transmitted. The received index  gives the gain in dB, , using the relations and quantization step defined in subclause 5.2.3.3.4. The valuesandgiven in subclause 5.2.3.3.4. The quantized gain, , is then given according to subclause 5.2.3.3.4.

6.1.1.2.3.3 Enhancement of background noise

The anti-swirling technique is applied in inactive periods, at 9.6 kb/s for NB signals, and 9.6 kb/s and below for WB and SWB signal. This technique is based on the decoded SAD and noisiness parameters. Basically, the anti-swirling effect is achieved by means of LP parameter smoothing in combination with reducing the power variations and spectral fluctuations of the excitation signal during detected periods of signal inactivity.

6.1.1.2.3.3.1 LP parameter smoothing

The LP parameter smoothing is done in two steps. First, a low-pass filtered set of LSP parameters is calculated by first-order autoregressive filtering according to

 ()

Hererepresents the low-pass filtered frame-end LSP parameter vector obtained for the current frame, is the decoded frame-end LSP parameter vector for the current frame, and is a weighting factor controlling the degree of smoothing.

In a second step, a weighted combination between the low-pass filtered LSP parameter vector, , and the decoded LSP parameter vectors, , and, is calculated using a weighting factor . That is

 (1460)

As mentioned in subclause 7.1.1, LSP interpolation is performed to obtain four LSP vectors, each for an individual subframe. This interpolation is based on: the decoded frame-end LSP parameter vector of the previous frame, the decoded mid-frame LSP parameter vector in the current frame and the decoded frame-end LSP parameter vector of the current frame. Subsequently, instead of using these parameters, their smoothed versions, given in equation () are employed.

It is noteworthy that the degree of smoothing is controlled by means of the control factor, which is described in subclause 6.1.1.2.3.3.3.

6.1.1.2.3.3.2 Modification of the excitation signal

One essential element of the anti-swirling technique is the reduction of power and spectrum fluctuations of the signal during periods of signal inactivity.

In the first step, tilt compensation of the excitation signal is performed with a first-order tilt compensation filter given as

 ()

The coefficientis calculated as

 ()

whereand are the zero-th and the first autocorrelation coefficients of the original excitation signal. The tilt compensation is carried out on a subframe basis.

In the second step, the spectral fluctuations of the excitation signal are further reduced by replacing a part of it with a white noise signal. To this end, first a properly scaled random sequence of unit variance is generated. This signal is then scaled by means of a gain factor, , in such a way that its power equals the smoothed power of the excitation signal. The gain factor, , is obtained by filtering the RMS value of the excitation signal, denoted as , on a frame-by-frame basis. That is

 ()

The noise is scaled by multiplying all its samples by the gain factor. Then, with some weighting factor, , the excitation signal, , is combined with the scaled noise signal, denoted as . This is done according to the following equation leading to the smoothed excitation signal:

 ()

It is noteworthy that the degree of excitation signal smoothing is controlled by means of the control factor, which is described in subclause 6.1.1.2.3.3.3.

6.1.1.2.3.3.3 Controlling the background noise smoothing

The anti-swirling method described in the clauses above is controlled by means of the control parametersandin response to the received SAD and noisiness parameters.

First, the received and decoded noisiness parameter steers an intermediate smoothing control parameter, , such that it is ensured that the degree of smoothing is only increased gradually up to a maximum degree that is indicated by the received parameter. Given the received noisiness parameter, , an intermediate parameter, , is set according to the following relation:

 ()

whereis the stored intermediate control parameter from the previous frame and is the step-size with which the smoothing control parameters are steered towardsas long as they are greater than. In case the current frame is erased (), is set to the intermediate control parameter of the previous frame, .

The SAD parameter activates the smoothing operation only when the received SAD flag, , indicates inactivity. However, in order to decrease the risk that smoothing is enabled during active signal periods, erroneously declared as inactive, the background noise smoothing is only enabled after a hangover period of 5 frames. Further, whenever the SAD declares a frame as active, the smoothing operation is disabled. In order to avoid adding a new hangover period after spurious SAD activation, no hangover is added if the detected activity period is less or equal to 3 frames.

In addition to this SAD-driven activation, for quality reasons, it is important to avoid the anti-swirling operation being turned on too abruptly. To this end, after each hang-over period, a phase in period of frames is applied, during which the smoothing operation is gradually steered from inactivate to fully enabled. Accordingly, for the n-th frame of the phase-in period, the smoothing control parametersandare calculated as follows:

 ()

For all other frames (during which the smoothing is activated) and.

It is noteworthy that phase-in periods are only inserted after hangover periods, i.e., not after spurious SAD activations of less than 3 frames.

##### 6.1.1.2.4 Reconstruction of the excitation in IC/UC mode at 9.6 kbps

6.1.1.2.4.1 Decoding of the innovative excitation

In IC and UC modes at 9.6 kbps the decoding the algebraic codebood ecitation is the same as described in n subclause 6.1.1.2.1.2.

At WB, an additional Gaussian noise excitation is generated as described in subclause 5.2.3.4.2.

6.1.1.2.4.2 Gains decoding

In NB, only the algebraic codeword gain is calculated as

 ()

The algebraic codebook excitation energy in dB, , is computed as in equation (). The quantized gain in dB is given by

 ()

The quantization index (6 bits) is retrieved directly from the bitstream (subclause 5.2.3.4.3.2)..

For WB the quantized algebraic codeword gainand Gaussian noise excitation gain are decoded. They are calculated respectively as

 ()

 ()

The quantized gain in dB is given by

 ()

The quantization index (5 bits) and (2 bits) are retrieved from the bitstream. The predicted algebraic codevector energy, , is decoded for the whole frame prior to calculating the algebraic codebook gain in each subframe (subclause 5.2.3.4.3.2).

6.1.1.2.4.4 Total excitation

The total excitation in each subframe is constructed by

 ()

where  and  are the pre-filtered algebraic codevector and the pre-filtered Gaussian noise excitation respectively.

Only the algebraic codevector  is used to update the contents of the adaptive codebook for the next frame.

The excitation signal, , is then post processed as described in subclause 6.1.1.3 to obtain the post-processed excitation signal , which is finally used as an input to the synthesis filter.

##### 6.1.1.2.5 Reconstruction of the excitation in GSC

In GSC mode, the attack flag is first decoded (subclause 5.1.13.5.3). Then, the number of subframe is decoded. To do so, if the bit rate is 13.2 kbit/s and the coding mode is INACTIVE, the first step is to decode 1 bit to verify if the coded frame is a SWB unvoiced frame which would implies 4 subframes. Otherwise when the number of subframe is less than 4, the noise level  as defined in subclause 5.2.3.5.4 is decoded. If the bitrate is 13.2 kbit/s, then a supplementary bit is decoded to determine if the number of subframe is 1 or 2, for lower bitrate the number of subframe is 1.

Then the cut off frequency (as defined in subclause 5.2.3.5.6) is decoded and if it is different from 0, the time domain contribution is decoded (subclause 5.2.3.5.2). When a time domain contribution exists, it is converted in frequency domain and low pass filtered using the decoded cut-off frequency as described in subclause 5.2.3.5.6, otherwise the time domain contribution is set to 0.

Then the frequency domain component is decoded starting the gain of sub bands as defined in subclause 5.2.3.5.7. The gain information is then used to determine the bit allocation, the number of bands and the order of the bands to be decoded by the PVQ as described in subclause 5.2.3.5.8. Then the PVQ is decoding the spectral difference and spectral dynamic control and noise filling is applied on the decoded vector as described in subclause 5.2.3.5.10. When the spectral difference vector is complete, the gain is applied and it is combined, in the frequency domain, with the temporal contribution as described in subclause 5.2.3.5.11. If the decoded frequency excitation meets the given condition, predict the un-decoded frequency excitation by the decoded frequency excitation as described in subclause 5.2.3.5.12. The complete excitation in the frequency domain is revert back to time domain using the inverse DCT as described in subclause 5.2.3.5.12 and then a pre-echo removal is applied as in subclause 5.2.3.5.13 to get the total excitation .

#### 6.1.1.3 Excitation post-processing

Before the synthesis, a post-processing of the excitation signal, , is performed to form the updated excitation signal, , as follows.

##### 6.1.1.3.1 Anti-sparseness processing

An adaptive anti-sparseness post-processing procedure is applied to the pre-filtered algebraic codevector, . This is to reduce the perceptual artefacts arising from the sparseness of algebraic codebook vectors having only a few non-zero samples per subframe. The anti-sparseness processing consists of circular convolution of the algebraic codevector with an impulse response by means of an FFT. Three pre-stored impulse responses are used and a selection number 0, 1or 2 and is set to select one of them. A value of 2 or greater corresponds to no modification; a value of 1 corresponds to medium modification and a value of 0 corresponds to strong modification. The selection of the impulse response is performed adaptively based on the decoded adaptive codebook gain, , coding mode and bit rate.

The following selection procedure is employed where is the algebraic codebook gain in the previous subframe, are current and 5 previous subframes' adaptive codebook gains and  is the previous selection number.

 ()

##### 6.1.1.3.2 Gain smoothing for noise enhancement

A nonlinear gain smoothing technique is applied to the algebraic codebook gain, , in order to enhance the excitation in noise. Based on the stability and voicing of the signal segment, the gain of the algebraic codebook vector is smoothed in order to reduce fluctuation in the energy of the excitation in case of stationary signals. This improves the performance in case of stationary background noise. The voicing factor is given by

 ()

with giving a measure of signal periodicity

 (1475)

whereand are the energies of the scaled pitch codevector and scaled algebraic codevector, respectively. Note that since the value of  is between –1 and 1, the value of is between 0 and 1. Note that the factor is related to the amount of "unvoicing" with a value of 0 for purely voiced segments and a value of 1 for purely unvoiced segments.

A stability factor is computed based on a distance measure between the adjacent LP filters. Here, the factor is related to the LSF distance measure. The LSF distance is given by

 ()

where in the present frame, calculated in subclause 7.1.1, andare the LSFs in the previous frame. The stability factor is given by

 ()

The LSF distance measure is smaller in case of stable signals. As the value of is inversely related to the LSF distance measure, then larger values of correspond to more stable signals. The gain smoothing factor,, is given by

 ()

The value of  approaches 1 for unvoiced and stable signals, which is the case of stationary background noise signals. For purely voiced signals, or for unstable signals, the value of approaches 0. An initial modified gain, , is computed by comparing the algebraic codebook gain, , to a threshold given by the initial modified gain from the previous subframe, . Ifis larger than or equal to , then is computed by decrementingby 1.5 dB, constrained by . Ifis smaller than , then is computed by incrementingby 1.5 dB, constrained by . Finally, the algebraic codebook gain is modified using the value of the smoothed gain as follows

 (1479)

##### 6.1.1.3.3 Pitch enhancer

A pitch enhancer scheme modifies the total excitation of voiced signals by filtering the algebraic codebook excitation through an innovation filter. The filter frequency response emphasizes the higher frequencies and reduces the energy of the low-frequency portion of the innovative codevector. The filter coefficients are related to the periodicity of the signal. Therefore, the pitch enhancer is not applied to excitation in UC at low bit rates, i.e. birates < 9600 kb/s.

A filter of the form

 ()

is used where  if Hz and  if Hz, with being a periodicity factor given in equation (). The filtered algebraic codebook vector in the current subframe is given by

 ()

where the out-of-subframe samples and are set to zero. The updated post-processed excitation is given by

 ()

The above procedure can be done in one step by updating the excitation as follows

 ()

whereis the modified algebraic codebook gain from equation ().

##### 6.1.1.3.4 Music post processing

In case of a sound signals coded with the GSC, a music enhancer scheme modifies the total excitation corresponding to the sound signal in such a way that the quantization noise inserted between spectral tones during the encoding/decoding process can be reduced. The music enhancer consists of converting the decoded excitation into frequency domain, computing a weighting mask for retrieving spectral information lost in the quantization noise, and modifying the frequency domain excitation by applying the weighting mask to increase the spectral dynamics, and converting the modified frequency domain excitation back to time domain.

The current frequency domain post processing achieves higher frequency resolution, without adding delay to the synthesis. A weighting mask is created based on the past spectrum energy and used to improve the efficiency of the inter-tone noise removal. To achieve this post processing without adding delay to the codec, a symmetric trapezoidal window is used. It is centred on the current frame where the window is flat, and extrapolation is used to create the future signal. The advantage of working on the excitation signal rather than on the synthesis signal is that any potential discontinuities introduced by the post processing are smoothed out by the subsequent application of the LP synthesis filter. The following text describes the implementation of the music post processing.

6.1.1.3.4.1 Excitation buffering and extrapolation

To increase the frequency resolution, a frequency transform longer than the frame length is used. To do so, a concatenated excitation vector  is created by concatenating the last 192 samples of the previous frame excitation*,* the decoded excitation of the current frame , and an extrapolation of 192 excitation samples of the future frame *.* This is described below where is the length of the past excitation as well as the length of the extrapolated excitation, and  is the frame length. These correspond to 192 and 256 samples respectively, giving the total length  samples:

 ()

The extrapolation of the future excitation samples  is computed by periodically extending the current frame excitation signal  using the decoded factional pitch of the last subframe of the current frame. Given the fractional resolution of the pitch lag, an upsampling of the current frame excitation is performed using a 35 samples long Hamming windowed sinc function.

6.1.1.3.4.2 Windowing and frequency transform

Prior to the time-to-frequency transform a windowing is performed on the concatenated excitation. The selected window  has a flat top corresponding to the current frame, and it decreases with the Hanning function to 0 at each end. The following equation represents the window used:

 ()

When applied to the concatenated excitation, an input to the frequency transform having a total length  samples () is obtained in the prototype. The windowed concatenated excitation  is centered on the current frame and is represented with the following equation:

 ()

During the frequency-domain post processing phase, the concatenated excitation is represented in a transform-domain using a type II DCT giving a resolution of 10Hz. The frequency representation of the concatenated and windowed time-domain CELP excitation *fu* is given below:

 ()

where , is the concatenated and windowed time-domain excitation and  is the length of the frequency transform. The frame length  is 256 samples, but the length of the frequency transform  is 640 samples for a corresponding inner sampling frequency of 12.8 kHz.

6.1.1.3.4.3 Energy per band and per bin analysis

After the DCT, the resulting spectrum is divided into critical frequency bands. The critical frequency bands used in the prototype are as close as possible to what is specified in [17], and their upper limits are defined as follows:

 ()

The 640-point DCT results in a frequency resolution of 10 Hz (6400Hz/640pts). The number of frequency bins per critical frequency band is

 ()

The average spectral energy per critical frequency band  is computed as follows:

 ()

where represents the *h*th frequency bin of a critical band and  is the index of the first bin in the *i*th critical band given by

 ()

The spectral analysis also computes the energy of the spectrum per frequency bin,  using the following relation:

 ()

Finally, the spectral analysis computes a total spectral energy of the concatenated excitation as the sum of the spectral energies of the first 17 critical frequency bands using the following relation:

 ()

6.1.1.3.4.4 Excitation type classification

The method for enhancing decoded generic sound signal includes an additional analysis of the excitation signal designed to maximize the efficiency of the inter-harmonic noise reducer by identifying which frame is well suited for the inter-tone noise reduction.

This classifier not only separates the decoded concatenated excitation into sound signal categories, but it also gives instructions to the inter-harmonic noise reducer regarding the maximum level of attenuation and the minimum frequency where the reduction can starts.

The first operation consists in performing an energy stability analysis based on the total spectral energy of the concatenated excitation:

 ()

where represents the average difference of the energies of the concatenated excitation vectors of two adjacent frames,  represents the energy of the concatenated excitation of the current frame *,* and  represents the energy of the concatenated excitation of the previous frame . The average is computed over the last 40 frames.

Then, a statistical deviation  of the energy variation over the last fifteen (15) frames is calculated using the following relation:

 ()

where, in the prototype, the scaling factor  is found experimentally and set to about 0.77. The resulting deviation  is compared to four (4) floating thresholds to determine to what extend the noise between harmonics can be reduced. The output of this second stage classifier is split into five (5) sound signal categories , named sound signal categories 0 to 4. Each sound signal category has its own inter-tone noise reduction tuning.

The five (5) sound signal categories 0-4 can be determined as indicated in the following table.

Table 156: Output characteristic of the excitation classifier

|  |  |  |
| --- | --- | --- |
| Category (eCAT) | Enhanced band (Hz) | Allowed reduction (dB) |
| 0 | NA | 0 |
| 1 | [510, 6400] | 6 |
| 2 | [510, 6400] | 9 |
| 3 | [400, 6400] | 12 |
| 4 | [300, 6400] | 12 |

The sound signal category 0 is a non-tonal, non-stable sound signal category which is not modified by the inter-tone noise reduction technique. This category of the decoded sound signal has the largest statistical deviation of the spectral energy variation and in general comprises speech signal.

Sound signal category 1 (largest statistical deviation of the spectral energy variation after category 0) is detected when the statistical deviation of spectral energy variation history is lower than Threshold 1 and the last detected sound signal category is ≥ 0. Then the maximum reduction of quantization noise of the decoded tonal excitation within the frequency band 510 to  Hz is limited to a maximum noise reduction  of 6 dB.

Sound signal category 2 is detected when the statistical deviation of spectral energy variation is lower than Threshold 2 and the last detected sound signal category is ≥ 1. Then the maximum reduction of quantization noise of the decoded tonal excitation within the frequency band 510 to  Hz is limited to a maximum of 9 dB.

Sound signal category 3 is detected when the statistical deviation of spectral energy variation is lower than Threshold 3 and the last detected sound signal category is ≥ 2. Then the maximum reduction of quantization noise of the decoded tonal excitation within the frequency band 4000 to  Hz is limited to a maximum of 12 dB.

Sound signal category 4 is detected when the statistical deviation of spectral energy variation is lower than Threshold 4 and when the last detected signal type category is ≥ 3. Then the maximum reduction of quantization noise of the decoded tonal excitation within the frequency band 300 to  Hz is limited to a maximum of 12 dB.

The floating thresholds 1-4 help preventing wrong signal type classification. Typically, decoded tonal sound signal representing music gets much lower statistical deviation of its spectral energy variation than speech. However, even music signal can contain higher statistical deviation segment, and similarly speech signal can contain segments with lower statistical deviation. It is nevertheless unlikely that speech and music contents change regularly from one to another on a frame basis. The floating thresholds add decision hysteresis and act as reinforcement of previous state to substantially prevent any misclassification that could result in a suboptimal performance of the inter-harmonic noise reducer.

Counters of consecutive frames of sound signal category 0, and counters of consecutive frames of sound signal category 3 or 4, are used to respectively decrease or increase the thresholds.

For example, if a counter counts a series of more than 30 frames of sound signal category 3 or 4, all the floating thresholds (1 to 4) are increased by a predefined value for the purpose of allowing more frames to be considered as sound signal category 4.

The inverse is also true with sound signal category 0. For example, if a series of more than 30 frames of sound signal category 0 is counted, all the floating thresholds (1 to 4) are decreased for the purpose of allowing more frames to be considered as sound signal category 0. All the floating thresholds 1-4 are limited to absolute maximum and minimum values to ensure that the signal classifier is not locked to a fixed category.

In the case of frame erasure, all the thresholds 1-4 are reset to their minimum values and the output of the signal classifier is considered as non-tonal (sound signal category 0) for three (3) consecutive frames (including the lost frame).

If information from a Voice Activity Detector (VAD) is available and it is indicating no voice activity (presence of silence), or if the last frame didn’t contain generic audio the decision of the signal type classifier is forced to sound signal category 0 ().

6.1.1.3.4.5 Inter-tone noise reduction in the excitation domain

Inter-tone noise reduction is performed on the frequency representation of the concatenated excitation as a first operation of the enhancement. The reduction of the inter-tone quantization noise is performed by scaling the spectrum in each critical band with a scaling gain  limited between a minimum and a maximum gain  and .The scaling gain is derived from an estimated signal-to-noise ratio (SNR) in that critical band. The processing is performed on frequency bin basis and not on critical band basis. Thus, the scaling gain is applied on all frequency bins, and it is derived from the SNR computed using the bin energy divided by an estimation of the noise energy of the critical band including that bin. This feature allows for preserving the energy at frequencies near harmonics or tones, thus substantially preventing distortion, while strongly reducing the noise between the harmonics. The inter-tone noise reduction is performed in a per bin manner over all 640 bins.

The minimum scaling gain  is derived from the maximum allowed inter-tone noise reduction in dB, . As described above, the second stage of classification makes the maximum allowed reduction varying between 6 and 12 dB. Thus minimum scaling gain is given by

 ()

The scaling gain is computed related to the SNR per bin. Then per bin noise reduction is performed on the entire spectrum to the maximum frequency of 6400 Hz. The noise reduction can start at the 2th critical band (i.e. no reduction is performed below 300Hz). The excitation type classifier module can push the starting critical band up to the 4th band (510 Hz), to reduce any potential degradation. This means that the first critical band on which the noise reduction is performed is between 300Hz and 920 Hz, and it can vary on a frame basis. In a more conservative implementation, the minimum band where the noise reduction starts can be set higher.

The scaling for a certain frequency bin  is computed as a function of , given by

, bounded by  ()

Usually  is equal to 1 (i.e. no amplification is allowed), then the values of  and  are determined such as  for dB, and  for dB. That is, for SNRs of 1 dB and lower, the scaling is limited to  and for SNRs of 45 dB and higher, no noise reduction is performed (). Thus, given these two end points, the values of  and  in equation are given by

 and  ()

If is set to a value higher than 1, then it allows the process to slightly amplify the tones having the highest energy. This can be used to compensate for the fact that the CELP codec, used in the prototype, doesn’t match perfectly the energy in the frequency domain. This is generally the case for signals different from voiced speech.

The SNR per bin in a certain critical band  is computed as

 ()

where  and  denote the energy per frequency bin for the past and the current frame spectral analysis, respectively, as computed in subclause 5.1.5.2, denotes the noise energy estimate of the critical band ,  is the index of the first bin in the *i*th critical band, and  is the number of bins in the critical band  as defined above.

The smoothing factor is adaptive and it is made inversely related to the gain itself. The smoothing factor is given by . That is, the smoothing is stronger for smaller gains . This approach substantially prevents distortion in high SNR segments preceded by low SNR frames, as it is the case for voiced onsets. The smoothing procedure is able to quickly adapt and to use lower scaling gains on onsets.

In case of per bin processing in a critical band with index , after determining the scaling gain and using  the actual scaling is performed using a smoothed scaling gain , updated in every frequency analysis as follows

 ()

Temporal smoothing of the gains substantially prevents audible energy oscillations while controlling the smoothing using  substantially prevents distortion in high SNR segments preceded by low SNR frames, as it is the case for voiced onsets or attacks.

The scaling in the critical band  is performed as

 ()

where  is the index of the first bin in the critical band  and  is the number of bins in that critical band.

The smoothed scaling gains  are initially set to 1. Each time a non-tonal sound frame is processed , the smoothed gain values are reset to 1 to reduce any possible reduction in the next frame.

Note that in every spectral analysis, the smoothed scaling gains  are updated for all frequency bins in the entire spectrum. Note that in case of low-energy signal, inter-tone noise reduction is limited to -1.25 dB. This happens when the maximum noise energy in all critical bands,  is less or equal to 10.

6.1.1.3.4.6 Inter-tone quantization noise estimation

The inter-tone quantization noise energy per critical frequency band is estimated as being the average energy of that critical frequency band excluding the maximum bin energy of the same band. The following formula summarizes the estimation of the quantization noise energy for a specific band :

 ()

where is the index of the first bin in the critical band ,  is the number of bins in that critical band*,*  is the average energy of a band ,  is the energy of a particular bin and *NB(i)* is the resulting estimated noise energy of a particular band. The variable represents a noise scaling factor per band that is found experimentally and is set such that more noise can be removed in low frequencies and less noise in high frequencies as it is shown below:

 ()

6.1.1.3.4.7 Increasing spectral dynamic of the excitation

The second operation of the frequency post processing provides an ability to retrieve frequency information that is lost within the coding noise. The CELP codecs, especially when used at low bitrates, are not very efficient to properly code frequency content above 3.5-4 kHz. The following steps take advantage of the fact that music spectrum does not often changed substantially from frame to frame. Therefore a long term averaging can be done and some of the coding noise can be eliminated. The following operations are performed to define a frequency-dependent gain function. This function is then used to further enhance the excitation before converting it back to the time domain.

6.1.1.3.4.8 Per bin normalization of the spectrum energy

The first operation consists in creating a weighting mask based on the normalized energy of the spectrum of the concatenated excitation. The normalization is done such that the tones have a value above 1.0 and the valleys a value under 1.0. To do so, the energy spectrum  is normalized between 0.925 and 1.925 to get the normalized energy spectrum  using the following equation:

 ()

where  represents the bin energy as calculated in subclause 5.1.5.2. Since the normalization is performed in the energy domain, many bins have very low values. The offset 0.925 has been chosen such that only a small part of the normalized energy bins would have a value below 1.0. Once the normalization is done, the resulting normalized energy spectrum is passed through a power function of 8 to obtain a scaled energy spectrum as shown in the following formula:

 ()

where  is the normalized energy spectrum and is the scaled energy spectrum. A maximum limit of the scaled energy spectrum is fixed to 5, creating a ratio of approximately 10 between the maximum and minimum normalized energy values. The following equation shows how the function is applied:

 ()

Where  represents limited scaled energy spectrum and is the scaled energy spectrum.

6.1.1.3.4.9 Smoothing of the scaled energy spectrum along the frequency axis and the time axis

With the last two operations, the position of the most energetic pulses begins to take shape. Applying power of 8 on the bins of the normalized energy spectrum is a first operation to create the mask that increases the spectral dynamics. The next 2 operations enhance this spectrum mask. First the scaled energy spectrum is smoothed along the frequency axis from low frequency to the high frequency with an averaging filter. Then, the resulting mask is processed along the time domain axis to smooth the bin values from frame to frame.

The smoothing of the scaled energy spectrum along the frequency axis can be described with following function:

 ()

Finally, the smoothing along time axis results in a time-averaged amplification/attenuation weighting mask  to be applied to the spectrum . The weighting mask, also called gain mask, is described with the following equation:

 ()

where  is the scaled energy spectrum smoothed along the frequency axis,  is the frame index, and  is the time-averaged weighting mask.

A slower adaptation rate has been chosen for the lower frequencies to substantially prevent gain oscillation. A faster adaptation rate is allowed for higher frequencies since the positions of the tones are more likely to change rapidly in the higher part of the spectrum. With the averaging performed on the frequency axis and the long term smoothing performed along the time axis, the final vector  is used as a weighting mask to be applied directly on the enhanced spectrum of the concatenated excitation.

6.1.1.3.4.10 Application of the weighting mask to the enhanced concatenated excitation spectrum

The weighting mask defined above is applied differently depending on the output of the excitation type classifier (value of ). The weighting mask is not applied if the excitation is classified as category 0 (; i.e. high probability of speech content).

For the first 1 kHz, the mask is applied if the excitation is not classified as category 0 (). Attenuation is possible but no amplification is performed in this frequency range (maximum value of the mask is limited to 1).

If more than 25 consecutive frames are classified as category 4 (; i.e. high probability of music content), but not more than 40 frames, then the weighting mask is applied without amplification for all the remaining bins (the maximum gain  is limited to 1, and there is no limitation on the minimum gain).

When more than 40 frames are classified as category 4, for the frequencies between 1 and 2 kHz the maximum gain  is set to 1.5 for bitrates below 12650 bits per second (b/s). Otherwise the maximum gain  is set to 1. In this frequency band, the prototype fixes the minimum gain  to 0.75 only if the bitrate is higher than 15850 b/s, otherwise there is no limitation on the minimum gain.

For the band 2 to 4 kHz, the maximum gain  is limited to 2 for bitrates below 12650 b/s, and it is limited to 1.25 for the bitrates equal to or higher than 12650 b/s and lower than 15850 b/s. Otherwise, then maximum gain  is limited to 1. Still in this frequency band, the minimum gain is 0.5 only if the bitrate is higher than 15850 b/s, otherwise there is no limitation on the minimum gain.

For the band 4 to 6.4 kHz, the maximum gain  is limited to 2 for bitrates below 15850 b/s and to 1.25 otherwise. In this frequency band, the prototype fixes the minimum gain  to 0.5 only if the bitrate is higher than 15850 b/s, otherwise there is no limitation on the minimum gain.

6.1.1.3.4.11 Inverse frequency transform and overwriting of the current excitation

After the frequency domain enhancement is completed, an inverse frequency-to-time transform is performed in order to get the enhanced temporal excitation back. The frequency-to-time conversion is achieved with the same type II DCT as used for the time-to-frequency conversion. The modified time-domain excitation  is obtained as

 ()

where is the frequency representation of the modified excitation,  is the enhanced concatenated excitation, and  is the length of the concatenated excitation vector.

To avoid adding delay to the synthesis, it has been decided to avoid overlap-and-add algorithm in the LP domain path. Thus, the exact length of the final excitation  is used to generate the synthesis directly from the enhanced concatenated excitation, without overlap as shown in the equation below:

 ()

Here  represents the length of the section of the window that was applied on the past segment of the excitation, prior to the frequency transformation.

### 6.1.2 Source Controlled VBR decoding

### 6.1.3 Synthesis

The LP synthesis is performed by filtering the post-processed excitation signal  through the LP synthesis filter*.*. The decoded and interpolated LP coefficients,, are used to construct the synthesis filter, .

The reconstructed speech for the subframe of size 64 is given by

 ()

The synthesized signal is then de-emphasized by filtering through the filter  (inverse of the pre-emphasis filter applied at the encoder input).

The de-emphasis synthesis speech  is then passed through an adaptive post-processing which is described in the following section.

### 6.1.4 Post-processing

The decoded signal is conveyed to several post-processing blocks. First an adaptive post-filtering is applied for enhancing the formant and harmonic structure of the signal. In a second step, a bass-post-filter treats the low frequencies.

6.1.4.1 Adaptive post-filtering

The post-filtering is similar to ITU-T G.729 post-processing with the main difference that it is performed at 12.8 or 16 kHz. The adaptive post-filter is a cascade of three filters: a long-term post-filter, , a short-term post-filter, , and a tilt compensation filter, , followed by an adaptive gain control procedure. The post-filter coefficients are updated in every subframe. The post-filtering process is organized as follows. First, the reconstructed signal, , is inverse-filtered through  to produce the residual signal, . This signal is used to compute the delay, , and gain, , of the long-term post-filter . The signal, , is then filtered through the long-term post-filter, , and the synthesis filter, . Finally, the output signal of the synthesis filter,  is passed through the tilt compensation filter, , to generate the post-filtered reconstructed signal, . Adaptive gain control is then applied to  to match its energy to the energy of . The post-filter parameters and are described in detail in subclauses 6.1.4.1.3. and 6.1.4.1.4.

The long-term post-filter is only applied for NB modes and is bypassed for WB and SWB. In WB and SWB cases, the post-filtering consists of a cascade of only two filters: a short-term post-filter,  (see subclause 6.1.4.3), and a tilt compensation filter,  (see subclause 6.1.4.4), followed by an adaptive gain control procedure (see subclause 6.1.4.5).

At 9.6 kbit/s NB decoding, the long-term post-filter, is active only for clean speech when the level of background noise is less than 20 dB. It is also desactivacted for UC mode.

##### 6.1.4.1.1 Long-term post-filter

The long-term post-filter is given by:

 ()

where is the pitch delay, and *gl* is the gain coefficient. Note that is bounded by 1, and is set to zero if the long-term prediction gain is less than 3 dB. The factor  controls the amount of long‑term post-filtering and has the value of . The long-term delay and the gain are computed from the residual signal, , obtained by filtering  through , which is the numerator of the short-term post-filter (see subclause 6.1.4.2). That is

 ()

The long-term delay is computed using a two-pass procedure. The first pass selects the best integer pitch delay, , in the range , where  is the integer part of the (transmitted) fractional pitch lag in the first subframe. The best integer, , is the one that maximizes the correlation

 ()

The second pass chooses the best fractional pitch delay, , with resolution 1/8 around . This is done by finding the delay with the highest pseudo-normalized correlation

 ()

where  is the residual signal at a fractional delay, . The fractional delayed signal, , is first computed using an interpolation filter of length 33. Once the optimal fractional delay, , is found,  is recomputed with a longer interpolation filter of length 129. The new signal replaces the previous one only if the longer filter increases the value of. Then, the corresponding correlation, , is normalized with the square-root of the energy of . The squared value of this normalized correlation is then used to determine if the long-term post-filter should be disabled. That is, if

 ()

the long-term post-filter is disabled by setting . Otherwise, the value of  is computed as

, constrained by  ()

##### 6.1.4.1.2 Short-term post-filter

The short-term post-filter is given by

 ()

where  is the quantized LP analysis filter (LP analysis is not done at the decoder) and the factors  and  control the amount of short-term post-filtering. The gain, , is calculated on the truncated impulse response, , of the filter  and is given by

 ()

Note that the gain, , will be modified according to the noise level as explained in the next clause.

##### 6.1.4.1.3 Post-filter NB parameters

In the ITU-T G.729 codec, the post-filter parameters ,and have fixed values. If a variable, called the long-term normalized noise gain, , is less than 25.0 and an active signal is detected,  has a value limited in the range [0.55, 0.70] and  has a value limited in the range [0.65, 0.75] as expressed by

 ()

 ()

Otherwise (not an active signal or  ≥ 25.0),  = 0.1 and  = 0.15.

In the case of the GSC mode the post-filter parameters , and are set to 1.

The long‑term normalized noise gain, , is updated only when in UC mode and when no signal activity is detected (). The update is performed as

 ()

where  is the normalized gain of random excitation in the UC mode, calculated as

 (1523)

In the equation above,  is the quantized gain of the random excitation, , used in TC mode, which has been quantized with 7 bits in the logarithmic energy domain. The modified value of in equation (1523) is not filtered. The modified value of is computed as

 ()

where the factor  is derived from  as follows

, constrained by  ()

Thus, the short-term post-filter, described in subclause 6.1.4.1.2, is used with the modified value of gain, , and not . These modifications help to diminish the effect of post-filtering in noisy conditions.

##### 6.1.4.1.4 Post-filter WB and SWB parameters

The post-filter parameters , for WB and SWB have fixed values, which depend on decoding mode. The filter may operate at both internal sampling frequencies 12.8 kHz and 16 kHz. In case of 12.8 kHz internal frequency the parameters take the default value  = 0.7, = 0.75

Table 157 Post filter WB and SWB parameters for 12.8 kHz

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Mode | Inactive & AMRWB IO clean speech | < 13.2 kbit/s clean speech | < 24.4 kbit/s clean speech | ≤ 32 kbit/s clean speech | < 15.85 kbit/s noisy speech | ≤ 32 kbit/s noisy speech |
|  | 0.7 | 0.80 | 0.75 | 0.72 | 0.75 | 0.7 |

In case of 16 kHz internal frequency, noisy speech (the level of background noise is less than 20) and for any mode not depicted in the table below the parameters take the default value  = 0.76, = 0.76.

Table 158 Post filter WB and SWB parameters for 16 kHz

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Mode | 13.2 kbit/s | 16.4 kbit/s | 24.4 kbit/s | 32 kbit/s |
|  | 0.82 | 0.80 | 0.78 | 0.78 |

##### 6.1.4.1.5 Tilt compensation

The filter compensates for the tilt in the short-term post-filter  and is given by

 ()

where  is a tilt factor with  being the first reflection coefficient, calculated fromas

 ()

where

 ()

The gain term compensates for the decreasing effect ofin. Furthermore, it has been shown that the product has generally no gain. Two values are used for  depending on the sign of . If  is negative, = 0.9, and if  is positive,  = 0.2.

##### 6.1.4.1.6 Adaptive gain control

Adaptive gain control is used to compensate for gain differences between the synthesized signal, , and the post-filtered signal, . A gain factor, , for the current subframe is computed by

 ()

Then, the post-filtered signal, , is scaled as

, for *n* = 0,…,63 ()

where ** is a continuous gain, updated on a sample-by-sample basis for NB input as

for NB input

, for n = 0,…,63 ()

for SWB TBE input

, for n = 0,…,63 ()

The initial value of  is used. Then, for each new subframe,  is set equal to  of the previous subframe.

For NB signals, the post-filtered synthesized signal, , is used instead of  for signal de‑emphasis, as described in subclause 6.3.

#### 6.1.4.2 Bass post-filter

This clause describes the functionality of the bass post-filter, a low-frequency pitch enhancement procedure, which is closely related to the corresponding procedures in [11].

The main difference compared to the previous standards is that the last step of post filtering is performed in the frequency domain. The reason for this is a different method of resampling from the internal sampling frequency to the output sampling frequency. Instead of time domain resampling (see clause 7.6 in [25]) complex low delay filter bank synthesis is used (see subclause 6.9).

The filter is applied to all LP-based modes up to 32 kbit/s except for NB noisy speech (the level of background noise > 20).

The bass post-filter uses two-band decomposition and adaptive filtering is applied only to the lower band. This results in a total post-processing that is mostly targeted at frequencies near the first harmonics of the synthesized signal.



Figure 91: Block diagram of bass post-filter

Figure shows the block diagram of the low-band pitch enhancer. Note that this is a simplified block diagram, which is equivalent to adding the low-pass filtered enhanced signal to the high-pass filtered signal (see subclause 6.1.3 in [11]). The decoded signal, , is first processed through an adaptive pitch enhancer module leading to an enhanced (full-band) signal, . By subtracting the decoded signal, an enhancement signal, , is obtained. Then CLDFB analysis (see subclause 5.1.2) is applied to transform signal into frequency domain . This signal is subsequently filtered in the frequency domain through a low-pass filter to obtain the signal  which is the low-band part of this response. The enhanced signal after post-processing, , is then obtained by adding the low-band enhancement signal to the transformed into frequency domain decoded signal. Resampling to the output sampling frequency and converting into time domain signal, , which is performed by CLDFB synthesis, is not a part of the bass post-filter and is applied for all modes (see subclause 6.9).

The object of the pitch enhancer module is to reduce the inter-harmonic noise in the decoded signal, which is achieved here by a time-varying linear filter described by the following equation:

 (1533)

where  is the output signal of the pitch enhancer, is a coefficient that controls the inter-harmonic attenuation. The signal  is the two-sided long-term prediction signal that is computed in each subframe as

 (1534)

where  is the pitch period of the decoded signal . Parameters  and  vary in time. With a value of , the gain of the filter described by equation () is exactly 0 at frequencies ,, , etc.; i.e., at the mid-point between the harmonic frequencies , , , , , etc. When approaches 0, the attenuation between the harmonics produced by the filter of equation () decreases.

The pitch lag parameter  is the received closed-loop pitch lag of the respective subframe. However, this parameter is only accurate for the part of the two-sided long-term prediction of Equation () predicting from the past pitch cycle. The prediction from the future pitch cycle may be less accurate, especially if the pitch lag value is not constant.

Thus, in order to improve the prediction accuracy, in case of voiced onset frames it is preferable to make use of the pitch lag value of the subframe containing the future pitch cycle, i.e., of that subframe whose closed-loop pitch lag points into the present subframe. This requires the availability of pitch lag parameters of a frame following the current frame.

The pitch lag parameter, , is further enhanced by means of a pitch tracker which makes the pitch contour smoother and avoids pitch doublings.

The factor  is computed as follows. First, the correlation between the signal and the predicted signal is given by

 ()

and the energy of the predicted signal is given by

 ()

The factor  is given by

, constrained by , ()

where  is the mean prediction error energy in dB in the present subframe and *k*1 takes values of 0.5 or 1 depending on the operating point. The mean prediction error energy, is updated as follows. The long-term prediction error is first computed by

 ()

where *k*2 equals *Cp*/*Ep* or 1 depending on the operating point, and then emphasized in the low frequencies using the relation

 ()

The energy of the emphasized error signal is then computed in dB as

 ()

and the mean error energy is then updated in every subframe by

 ()

with initial value .

The factor  is further adapted to a measure of signal stationarity, which limits the level of inter‑harmonic attenuation when the signal is not in a steady-state mode. The adaptation is based on the stability factor of the current frame,  and a recursively smoothed version of stability factor  defined as

 ()

The factor , defined in equation (1537), is finally scaled as

 ()

Since larger portions of noise are aurally masked when the signal rapidly changes, the above adaptation gives a better balance between attenuation of quantization noise and signal degradation.

At 16.4 and 24.4 kbps, the factor  is adjusted by decoding the gain adjustment , which is quantized at the encoder (see subclause 5.2.4) and transmitted in the bitstream on 2 bits.

 ()

### 6.1.5 Decoding of upper band for LP-based Coding Modes

#### 6.1.5.1 Decoding Time-domain Bandwidth Extension

The time domain bandwidth extension decoder synthesizes the high band excitation signal from the excitation signal generated by the low band ACELP decoder and a set of high band model parameters received from the time domain bandwidth extension encoder. The synthesized high band signal is then combined with the output from the lowband ACELP decoder to generate a superwideband output. The high level schematic of the time domain bandwidth extension decoder is shown in figure 92.



Figure 92: Time domain bandwidth extension decoder

##### 6.1.5.1.1 Generation of the upsampled version of the lowband excitation

An upsampled version of the low band excitation signal is derived from the ACELP core as show in figure . For each ACELP core coding subframe, *i*, a random noise scaled by a factor voice factor,  is first added to the fixed codebook excitation that is generated by the ACELP core encoder. The voice factor is determined using the subframe maximum normalized correlation parameter,  that is derived during the ACELP encoding. First the  factors are combined to generate.

 (1545)

 calculated above is limited to a maximum of 1 and a minimum of 0. When the ACELP coder type is voiced the Vf is modified based on the integer pitch value T0 is modified as in the pseudo-code below:

if((coder\_type == VOICED))

if(T0 <= 57.75f)

 = -0.0126f\*T0 + 1.23f;

else if(T0 > 57.75f && T0 < 115.5f)

 = 0.0087f\*T0;

end

end

Regardless of the ACELP coder type, if the open loop pitch lag T0 exceeded 115, Vf is set to 1;

 if the ACELP core encodes a maximum of 6.4 KHz or  if the ACELP core encodes a maximum bandwidth of 8 KHz.

The ACELP fixed code book excitation signal mixed with noise is then resampled by a factor . The resampling factor is set to 5/2 when the ACELP core encodes a maximum bandwidth of 6.4 KHz and it is set to 2 when the ACELP core encodes a maximum bandwidth of 8 KHz.

The resampled output is scaled by the ACELP fixed codebook gain and added to a delayed version of itself.

 (1546)

where gc is the subframe ACELP fixed codebook gain, gp is the subframe ACELP adaptive codebook gain and P is the open loop pitch lag.

##### 6.1.5.1.2 Non-Linear Excitation Generation

The excitation signal  is processed through a non-linear function in order to extend the pitch harmonics in the low band signal into the high band. The non-linear processing is applied to a frame of  in two stages; the first stage works on the first half subframe (160 samples) of  and the second stage works on the second half subframe. The non-linear processing steps for the two stages are described below. In the first stage , and in the second stage, .

First, the maximum amplitude sample  and its location relative to the first sample in the stage  are determined.

 (1547)

Based on the value of , the scale factor  is determined.

 (1548)

The scale factor  and the previous scale factor parameter from the memory  are then used to determine the parameter scale step.

 (1549)

If , then 

The output of the non-linear processing  is derived as per

 (1550)

The previous scale factor parameter is updated recursively for all  according to

for(j=n1; j< n2; j++)

if(j<imax)

=

end

end

##### 6.1.5.1.3 De-quantization of high band parameters

The high band LSF, gain shape and gain frame parameters are de-quantized by looking up the quantization tables for these parameters based on the indices. The LSF de-quantization is done as follows:

6.1.5.1.3.1 LSF de-quantizing

The first five LSFs  are reconstructed directly from the received CB indices. The mirroring frequency and optimal grid are reconstructed from the received indics. The upper-half ot the coefficients  are reconstructed by flipping the lower-half of the coefficients over the reconstructed mirroring frequency, rescaling and then smoothing with the reconstructed optimal grid, as described in subclause 5.2.4.1.3.1.

 (1551)

Using the received VQ index parameter for the gain shape, the de-quantized gain shape vector that contains the gain shape parameter in the log domain is retrieved. The quantized gain shape parameters  are then obtained from the log domain values by inverse logarithm operation.

The de-quantized frame gain parameter  is obtained by obtaining the log domain frame gain value from the table and by converting this back into linear domain by inverse logarithm operation.

For the bit rates of 24.4 kb/s and 32 kb/s, the de-quantized high band subframe residual energy , the de-quantized high band target energy, and the mixing factor,  ,are obtained by table lookup using the respective received indices.

##### 6.1.5.1.4 LSP interpolation

Refer to subclauses 5.2.6.1.4 and 5.2.6.1.4.2 for 24.4 kbps and 32kbps LSP interpolation.

6.1.5.1.4.1 LSP interpolation at 13.2 kbps and 16.4 kbps

Refer to subclause 5.2.4.1.4.1

##### 6.1.5.1.5 Spectral flip in time domain

The non-linear excitation  is spectrally flipped so that the high band portion of the excitation is modulated down to the low frequency region. This spectral flip is accomplished in time domain

 (1552)

##### 6.1.5.1.6 Down-sample using all-pass filters

 is then decimated using a pair of all pass filters to obtain an 8 KHz bandwidth (16 KHz sampled) excitation singal . This is done by filtering the even samples of  by an all pass filter whose transfer function is given by

 (1553)

And the odd samples of  by an all pass filter whose transfer function is given by

 (1554)

The 16 KHz sampled excitation signal  are obtained by averaging the outputs of the above filter.

These filter coefficients are described in subclause 5.2.6.1.9.

##### 6.1.5.1.7 Adaptive spectral whitening

Due to the nonlinear processing applied to obtain the excitation signal , the spectrum of this excitation is no longer flat. In order to flatten the spectrum of the excitation signal , 4th order linear prediction coefficients are estimated from The spectrum of  is then flattened by inverse filtering  using the linear prediction filter.

The first step in the adaptive whitening process is to estimate the autocorrelation of the excitation signal

 (1555)

A bandwidth expansion is applied to the autocorrelation coefficients by multiplying the coefficients by the expansion function:

 (1556)

The bandwidth expanded autocorrelation coefficients are used to obtain LP filter coefficients,  by solving the following set of equations using the Levinson-Durbin algorithm as described in section.

 (1557)

It must be noted that .

The whitened excitation signal  is obtained from  by inverse filtering

 (1558)

4 samples of  from the previous frame are used as memory for the above filtering operation.

For bit rates 24.4 kb/s and 32 kb/s, the whitened excitation is further modulated by the normalized residual energy parameter . In other words, for bitrates 24.4 kb/s and 32 kb/s,

 (1559)

##### 6.1.5.1.8 Envelope modulated noise mixing

To the whitened excitation, a random noise vector whose amplitude has been modulated by the envelope of the whitened excitation is mixed using a mixing ratio that is dependent on the extent of voicing in the low band.

First,  is calculated and then the envelope of the envelope of the whitened excitation signal is calculated by smoothing 

 (1560)

In SWB, the factors  and  are calculated using the voicing factors,  for subframes , determined from the low band ACELP encoder. The average of the 4 voicing factors, , is calculated and modified as . This is then confined to values between 0.6 and 0.999. Then  and  are estimated as

 (1561)

 (1562)

However, for bit rates 16.4 kb/s and 24.4 kb/s and if TBE was not used in the previous frame,  and  are set to

 (1563)

 (1564)

and for , is substituted by an approximated value  as

 (1565)

In WB mode, the factors  and are initialized to and. However, if the bitrate is 9.6kb/s, they might get reset to andif the if the low band coder type is voiced or, or to and if the low band coder type is unvoiced or.

A vector of random numbers,  of length 160 is then modulated by  to generate  as

 (1566)

The whitened excitation is then de-emphasized with  which is the pre-emphasised effect since the used spectrum is flipped.

 (1567)

If the lowband coder type is unvoiced, the excitation  is first rescaled to match the energy level of the whitened excitation 

 (1568)

where and then pre-emphasised with =0.68 to generate the final excitation which is the de-emphasised effect since the used spectrum is flipped.

 (1569)

If the lowband coder type was not un-voiced, the final excitation is calculated as

 (1570)

for each sample index within subframe .

For bit rates less than 24.4 kb/s, the mixing parameters  and  are estimated as

 (1571)

 (1572)

For bit rates 24.4 kb/s and 32 kb/s, the mixing parameters ,  are estimated as follows:

 (1573)

 (1573a)

where the parameter  is defined in in subclause 5.2.6.1.13.

 is then de-emphasised to generate the final excitation.

##### 6.1.5.1.9 Spectral shaping of the noise added excitation

The excitation signal  is then put through the high band LPC synthesis filter that is derived from the quantized LPC coefficients (see subclause 5.2.4.1.3).

For bitrates below 24.4 kb/s, a single LPC synthesis filter  is used and the shaped excitation signal  is generated as

 (1574)

For bitrates above 24.4 kb/s the LPC synthesis filter is applied to the excitation signal  in four subframes based on

 (1575)

##### 6.1.5.1.10 Post processing of the shaped excitation

Refer to subclause 5.2.6.1.13.

##### 6.1.5.1.11 Gain shape update

The gain shapes are updated according to the coding type of both the current frame and the previous frame. The pitch gain of the current frame is also taken into account.

The pitch gain of the current frame is calculated by:

 ()

If the coding type of the current frame and the previous frame are the same, or the coding type of the current frame is GENERIC and the coding type of the previous frame is VOICED, or the coding type of the current frame is VOICED and the coding type of the previous frame is GENERIC, and the pitch gain of the current frame  is greater than 70, then the gain shape parameters are smoothed as follows:

 ()

 ()

where are the subframe energies in the shaped excitation signal of the current frame,  are the subframe energies in the shaped excitation signal of the previous frameand are the quantized gain shape parameters of the previous frame.

##### 6.1.5.1.12 SHB synthesis

In order to smooth the evolution of the post-processed spectrally shaped highband excitation signal at the frame boundary, the energy ratio between the current frame’s overlap samples and the previous frame’s overlap samples are calculated as below:

 (1579)

where  is 20 samples. The tenth-order LPC synthesis performed as described according to subclause 6.1.5.1.9 uses a memory of ten samples, thus there is atleast a ten sample energy propagation from the previous frame into the current frame. When calculating the energy scaling to be applied to the current frame, it should be noted that the first 10 samples of the present frame are considered as a part of previous frame energy. If the voicing factor is greater than 0.75, the numerator in the above equation is attenuated by 0.25. The spectrally shaped high band excitation signal is then modified by the above scaling factor as follows:

 (1580)

For bit rates 24.4 kb/s or higher, gain shape values are then up sampled from 4 values to 16 values as described in below. First subframe energies from the spectrally shaped highband excitation signal are calculated.

 for  ()

The interpolated gain shape parameters are obtained as follows

 for 



Based on either  or (depending on the bit rate), the shaped highband excitation signal is scaled. The scaling is performed using overlapping windows as described below:

where the definition of swin1 is described in section 5.2.6.1.15. The scaled excitation is then finally multiplied by the quantized frame gain to obtain the highband synthesized signal.

 for *n=0,…359*

The overlap portion from the previous frame are then added to the first 20 samples of the current frame of .

The highband synthesized signal is then used to generate a 32 KHz sampled highband component of the final output decoded speech signal. First the highband synthesized signal is upsampled by 2 using an interpolator. The  signal is filtered through all-pass filters as per below.



and



Table 159: All-pass filter coefficients for interpolation by a factor of 2

|  | All pass coefficients |
| --- | --- |
| b0,1 | 0.06056541924291 |
| b1,1 | 0.42943401549235 |
| b2,1 | 0.80873048306552 |
| b0,2 | 0.22063024829630 |
| b1,2 | 0.63593943961708 |
| b2,2 | 0.94151583095682 |

And the output samples from both these filters are interlaced to generate the upsampled highband signal. At bitrate of 13.2 and 16.4 kb/s where the 12.8 KHz sampled core is used, the upsampled higband signal is downmixed using a Hilbert operator.

##### 6.1.5.1.13 Core-switching and high-band memory updates

6.1.5.1.13.1 TBE/IGF switching

When switching from ACELP to TCX core and thus from TBE to IGF, or vice versa, the transition of the high-band signals is performed implicitly by the cross-fade transition mechanism of the core signals. Due to differing delay compensations, the high-band IGF and TBE signals overlap, when switching from IGF to TBE. On the other hand, when switching from TBE to IGF the differing delay compensations cause a gap in between the high-band signals. To fill this gap and additionally provide overlapping signals for the cross-fade, a transition signal is generated as follows.

To obtain a continuous high-band signal to the previous frame, the overlap portion of the high-band synthesized signal  is used, as described in the SHB synthesis subclause 6.1.5.1.12. This overlap portion  is up-sampled using the same filters and. The output samples of the filters are interlaced, if the underlying core is sampled at 16 kHz or processed using a Hilbert operator to generate the up-sampled high-band signal.

The needed length of is 148 samples, so the 40 samples of are extrapolated using the temporally mirrored end of the high-band synthesized signal of the previous frame, where

 ()

The signals  and are merged to generate by overlap and add using the window as described in table , as

 ()

Table 160 Window for generation of transition signal

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| n |  | n |  | n |  | n |  |
| 1 | 0.04618346 | 11 | 0.4866045 | 21 | 0.8249975 | 31 | 0.9904104 |
| 2 | 0.09216993 | 12 | 0.5258704 | 22 | 0.8493099 | 32 | 0.9946717 |
| 3 | 0.1381564 | 13 | 0.5651364 | 23 | 0.8736224 | 33 | 0.9989330 |
| 4 | 0.1835534 | 14 | 0.6019916 | 24 | 0.8942082 | 34 | 0.9994665 |
| 5 | 0.2289505 | 15 | 0.6388468 | 25 | 0.9147939 | 35 | 1 |
| 6 | 0.2733710 | 16 | 0.6729768 | 26 | 0.9314773 | 36 | 1 |
| 7 | 0.3177914 | 17 | 0.7071068 | 27 | 0.9481606 | 37 | 1 |
| 8 | 0.3608562 | 18 | 0.7382205 | 28 | 0.9607993 | 38 | 1 |
| 9 | 0.4039210 | 19 | 0.7693340 | 29 | 0.9734381 | 39 | 1 |
| 10 | 0.4452628 | 20 | 0.7971658 | 30 | 0.9819242 | 40 | 1 |

##### 6.1.5.1.14 TEC/TFA post processing

The TEC and the TFA are complementally activated according to the transmitted information. The TEC is performed when an onset is detected in the high frequency band at the encoder (i.e. ). On the other hand, the TFA is performed when the temporal envelope of the high band signal is detected to be flat at the encoder (i.e. ).

Decoding the transmitted codeword, the TEC and TFA parameters are set as:

Table 161: Decoding the transmitted codeword to the TEC and TFA parameters.

|  |  |  |
| --- | --- | --- |
| Codeword | tec\_flag | tfa\_flag |
| 00 | 0 | 0 |
| 01 | 0 | 1 |
| 10 | 1 | 0 |
| 11 | 2 | 0 |

When , the temporal envelope of the high frequency band is calculated from the temporal envelope of the low frequency band and the TEC parameter and then the high frequency band signal is shaped with the calculated temporal envelope of the high frequency band. Firstly, the temporal envelope of the low frequency band of the decoded signal  is calculated as:

 ()

where

 ()

and where  is the low frequency band signal in the QMF domain described in subclause 6.1.4.2.

Then, the temporal envelope of the high frequency band is calculated from the temporal envelope of the low frequency band and the TEC parameter as:

 ()

where .

The gain values to be applied to the high frequency band signal is calculated as

 ()

The gain values are limited by the lower limit:

 ()

The lower limit  is defined as:

 ()

where

 ()

and where

 ()

Then, the gain values are modified for maintaining the energy of the high frequency band signal of the frame

 ()

Finally, the gain values are applied to the high frequency band signal

 ()

where is the subframe length of TEC and TFA which is 1.25 ms at the output sampling rate ().

When the , the temporal envelope of the high frequency band signal is determined as “flat” and then it is flattened as follows. The gain values for the TFA  are calculated by:

 ()

By applying the gain values, the temporal envelope of the high frequency band signal is flattened:

 ()

##### 6.1.5.1.15 Full-band synthesis

Four bits are decoded from the bitstream to obtain the energy ratio, and then calculate the  described as follows:

 ()

Interpolate the signal  (see subclause 5.2.6.1.17) from 16 kHz to 48 kHz with zeros

 ()

The interpolated signal  passes through the bandpass filter and gets the signal. The calculation of the energy  is described as follows:

 ()

The energy of is calculated as follows,

 ()

The synthesized full-band signal is calculated as follows,

 ()

#### 6.1.5.2 Multi-mode FD Bandwidth Extension decoding

The super higher band (SHB) signal for SWB signal or the higher band (HB) signal for WB signal is adaptively decoded with multi-mode BWE algorithm according to the result of the classification decision process of the SHB or HB signal decoded from the received bitstream and a determined excitation signal. In case of FB mode, the energy ratio of the current frame is decoded, the full-band (FB) signal is synthesized based upon the energy ratio or the envelope ratio calculated from low band envelope and the generated SHB frequency excitations. Combining with the low band signal decoded based on the received bitstream, the output signal is obtained.

##### 6.1.5.2.1 SWB multi-mode FD BWE decoding

First of all, the SHB signal class of current frame is decoded. Then the spectral envelopes or spectral/time envelopes are adaptively decoded depending upon the decoded SHB signal class. Four spectral envelopes and four time envelopes are decoded from the received bitstream for TRANSIENT frames. For all of the other cases, i.e. NON-TRANSIENT frames, fourteen spectral envelopes are decoded from the received bitstream and no time envelope is decoded. Frequency excitations are then generated according to the SHB signal class and finally, the super higher band signal is synthesised based upon the signal class, the decoded envelopes and generated frequency excitations.

6.1.5.2.1.1 Decoding the multi-mode FD BWE signal class

Two bits are decoded from the bitstream to obtain the SHB signal class according to subclause 5.2.6.2.1.3.

6.1.5.2.1.2 Decoding the spectral envelope

In TRANSIENT frames, the envelope VQ indices  are used to regenerate the synthesised signal envelope,

 (1601)

In Non-TRANSIENT frames, the envelope VQ indices  are used to generate the synthesised signal envelope,

(1602)

The final de-quantized envelope is then calculated:

 (1603)

where ,

 (1604)

6.1.5.2.1.3 Decoding the time envelope

If the current frame is a TRANSIENT frame, then four bits are decoded to obtain the index of each time envelope, . This envelope is converted into the linear domain as follows:

 ()

The linear domain time envelope of the previous sub-frame is preserved as. is set to zero for the first frame. Time envelope de-normalization is performed after the frequency domain processing in subclause 6.1.5.2.1.6.

6.1.5.2.1.4 Windowing and time-to-frequency transformation

640-point length MDCT is used for SWB FD BWE. Refer to subclause 5.3.2.

6.1.5.2.1.5 Frequency excitation generation

The base frequency excitation signal  is generated from the wideband MDCT coefficients  of synthesized wideband signal or from random noise depending on the decoded SHB signal class.

To Non-TRANSIENT frames, four parameters,  ,, , and spectral tilt of WB signal are calculated. When  and , if one of the condition: , *,* is satisfied, the fricative flag  is set to 1. It is noted that parameter initialized to 0. It is calculated for every frame and preserved as.

– The base frequency excitation coefficients are obtained from the wideband MDCT coefficients:

 (1606)

where is the cut-off spectrum bin of WB signal, and  at 13.2kbps and at 32kbps .

– If the current frame is a NOISE frame or  is equal to 1, the base frequency excitation signal is generated from linear congruential uniform random noise generator as follows

 (1607)

where

 (1608)

Parameter  is initialized as 21211 and updated for each MDCT coefficient. It should be noted that  is calculated for every frame.

– Otherwise, the base frequency excitation signal is copied as defined in subclause 5.2.6.2.1.5 (Frequency mapping to generate base excitation spectrum in FD BWE).

6.1.5.2.1.6 Frequency excitation normalization and spectral envelope de-normalization

Firstly, the base frequency excitation signal or the spectral envelope is adjusted depending upon the SHB signal class. Then the base frequency excitation signal is adaptively normalized to remove the original low frequency envelope information. Finally, the spectral envelopes are applied to the normalized excitation signal.

If the current frame is NORMAL and the fricative flag  (as defined in subclause 6.1.5.2.1.5) is equal to 0, the spectral envelope and the base frequency excitation signal are firstly adjusted. The spectral envelope is adjusted as follows:

 (1609)

where .

And the base frequency excitation signal is adjusted.

– while weighting factor  is smaller than 1, the base frequency excitation  is adjusted by multiplying by ,  is then increased by 0.1 and the index is then incremented by 1.

where, weighting factor  is initialized as follows:

 (1610)

where  is defined in subclause 5.2.6.2.1.5.

– while weighting factor  is larger than 1, the base frequency excitation  is adjusted by multiplying by ,  is then decreased by 0.5 and the index  is then decremented by 1.

where, weighting factor  is initialized as follows:

 (1611)

– while weighting factor  is larger than 1, the base frequency excitation  is adjusted by multiplying by ,  is then increased by 0.1 and the index  is then incremented by 1.

where, weighting factor is initialized as follows:

 (1612)

and  is defined in subclause 5.2.6.2.1.5.

– while weighting factor  is larger than 1, the base frequency excitation  is adjusted by multiplying by , and then  is multiplied by 0.95 and the index  is decremented by 1.

where, weighting factor  is initialized as follows:

 (1613)

Otherwise, there is no need to adjust the base frequency signal.

 (1614)

In order to normalize the base frequency excitation, the parameter of adaptive normalization length  is calculated depending on the decoded SHB signal class and the wideband MDCT coefficients:

– The 256 wideband MDCT coefficients in the 0-6400 Hz frequency range,  are split into 16 sharpness bands (16 coefficients per band). In sharpness band *j*, if  and , the counter  is incremented by one.

where , and the maximum magnitude of the spectral coefficients in a sharpness band, denoted , is:

 (1615)

Parameter  is initialized to 0 and calculated for every frame.

– Then the normalization length  is obtained:

 (1616)

where the current normalization length  is calculated depending on the SHB signal class:

 (1617)

and the current normalization length is preserved as .

When the current frame is not NOISE and the fricative flag  is equal to 0, the noise content of the base frequency excitation signal should be generated, and the base frequency excitation signal should be normalized to remove the core envelope information. The above algorithm is according to the adaptive normalization length .

The normalization envelopes are firstly calculated:

 (1618)

Then the signs and the amplitudes of the base frequency excitation signal are calculated by:

 (1619)

 (1620)

The adjusted coefficients are obtained by the amplitudes, the normalization envelopes and adaptive normalization length:

 (1621)

If , the adjusted coefficients are further modified by the modification factor , and then the base frequency excitation signal with the noise content is obtained by the signs and the adjusted coefficients:

 (1622)

 is the modification factor and can be determined as

 (1623)

where,  for HARMONIC frame, otherwise, , and  is preserved for the next frame.

The normalized frequency excitation is obtained by removing the core envelope information:

 (1624)

where .

Finally, the spectral envelope is applied to the normalized excitation signal to obtain the SHB coefficients.

– For TRANSIENT frames, it is achieved as follows:

 (1625)

where  and .

– For non-TRANSIENT frames, if  *AND* ,  and , the frequency signal is multiplied by 0.2:

 (1626)

Otherwise, there is no adjustment.

 (1627)

Then the MDCT coefficients of the reconstructed SHB signal are further adaptively adjusted with different modes:

 (1628)

where  for NOISE or NORMAL frame and for HARMONIC frame,  and the index  is incremented by  if  is smaller than 14.

The weighted envelopes  are obtained from the spectral envelopes of the current frame and the previous frame:

 (1629)

where,

 (1630)

and is calculated for every frame and preserved as the previous energyfor the next frame.

The index *k* is initialized to 0, and four weighting factors, , , ,  , are calculated. The smoothing factor is defined in table 162.

Table 162: Smoothing factor 

|  |  |
| --- | --- |
| *j* | Smoothing factor |
| 0 | 0.05 |
| 1 | 0.05 |
| 2 | 0.05 |
| 3 | 0.05 |
| 4 | 0.05 |
| 5 | 0.05 |
| 6 | 0.05 |
| 7 | 0.0417 |
| 8 | 0.0417 |
| 9 | 0.0417 |
| 10 | 0.0417 |
| 11 | 0.03125 |
| 12 | 0.03125 |

While the index *k* is smaller than 8, the frequency excitation  is adjusted by multiplying , and then the index *k* is incremented by 1, and  is increased by adding .

In the sub-band , while the index *k* is smaller than , the frequency excitation  is adjusted by multiplying by , and then, the index *k* is incremented by 1, and  is increased by adding .

In the 13th sub-band, while the *k* is smaller than , the frequency excitation  is adjusted by multiplying by, and then, the index *k* is incremented by 1.

The frequency excitation  is adjusted by the weighted spectral envelopes to obtain the final SHB frequency signal ,.

It should be noted that, for Non-TRANSIENT frames, the spectral envelopes of current frame are preserved as . For TRANSIENT frame, the spectral envelope is calculated and preserved:

 (1631)

Further adjustment is performed by:

 (1632)

6.1.5.2.1.7 Windowing and frequency-to-time transformation

640-point length inverse MDCT is used for SWB FD BWE. Refer to subclause 6.2.4.

6.1.5.2.1.8 Time domain post-processing

The SHB synthesis signal is adjusted depending upon the SHB class.

– If the current frame is TRANSIENT, the SHB synthesis signal in time domain is adjusted by the time envelopes to match the transient characteristics of the original signal.

The 640 SHB synthesized samples  are divided into 4 sub-frames, and the energy of each sub-frame is calculated:

 ()

next, the time envelope is adjusted:

 ()

and finally, the SHB synthesis signal is adjusted as follows:

 ()

where, and .

In this case, the value of is preserved for the next frame.

 ()

– Else if fricative flag defined in subclause 6.1.5.2.1.5 is equal to 1 and the previous fricative flagis equal to 0, pre-echo reduction is performed and the preserved time-domain energy is updated.

Firstly, the 640 ACELP core synthesized samples,  are divided into 4 sub-frames, and the energy of each sub-frame is calculated:

 ()

In *j*th sub-frame, if and , the position is introduced to separate 4 sub-frames into two parts and it is initialized as 0:  ()

Next, if , the SHB synthesis signal is adjusted.

 ()

where,  and  are the energies of the SHB synthesized samples  and the energy of SHB synthesized samples .  is the energy value from the previous frame, and if, the value of  is updated to .

– Otherwise the energy is calculated and preserved for next frame:

 ()

##### 6.1.5.2.2 WB multi-mode FD BWE decoding

The HB signal class and the spectral envelopes are decoded (at 13.2kbps) or predicted (at 7.2/8kbps), and the frequency excitations are generated from the decoded low-band synthesized signal or from random noise, and then the frequency excitations are adjusted along with the signal class and decoded or predicted spectral envelopes to obtain the higher band signal.

6.1.5.2.2.1 Decoding the multi-mode FD BWE signal class

At 13.2kbps, one bit is decoded from bitstream to get the HB signal class according to subclause 5.2.6.2.2.2. And at 7.2kbps or 8kbps, the HB signal class is set to NORMAL.

6.1.5.2.2.2 Windowing and time-to-frequency transformation

320-point MDCT is used for WB FD BWE. Refer to subclause 5.3.2.

6.1.5.2.2.3 Decoding the spectral envelope

At 13.2kbps, five bits are decoded to obtain the index of spectral envelope, . This envelope is converted into the linear domain as follows:

 (1641)

where  is defined in subclause 5.2.6.2.2.3.

The average BWE signal envelope of the current frame at 13.2kbps is preserved for the envelope estimation for 7.2kbps and 8kbps when bit-rate switching from 13.2kbps to 7.2 or 8kbps, as described as follows:

 (1642)

At 7.2kbps or 8kbps, the spectral envelope is predicted in the decoder. If the extended layer of the previous frame is different from the one of current frame, that is, , the preserved spectral envelope  are set to 0. In order to get the predicted spectral envelope, two bands andare firstly selected. Then three average energies are needed based on the frequency coefficients in the above two bands. Finally, the spectral envelopes used for the following frequency adjustment are obtained. In order to decrease the complexity, the energies are calculated with the MDCT coefficients :

 (1643)

 (1644)

 (1645)

Two factors, , are calculated. When  and , the energy variation flag  is set to 1. It should be noted that  is initialized to 0 and calculated for every frame.

Then the weighting factor is initialized to 1, and updated according to the spectral envelope and code type:

 (1646)

and the spectral envelope is accordingly adjusted by:

 (1647)

 (1648)

Next, the first envelope is further adjusted.

– The first envelope is firstly adjusted with by:

 (1649)

– If the conditions: ,, and **, are all satisfied, the first envelope is adjusted by:

 (1650)

Otherwise, there is no adjustment.

 (1651)

– If the conditions: , , , , , , are all satisfied, the first envelope is adjusted as follows:

 (1652)

and

 (1653)

where, the adjustment flagis initialized to 0. If the coder type of current frame is equal to that of the previous frame and the first spectral envelope is larger than the envelope of the previous frame, that is, , the adjustment flag  is set to 1. The values are the spectral envelopes of the previous frame.

Otherwise, there is no adjustment.

 (1654)

– If the conditions: , , , are all satisfied, the envelope variation flag  initialized to 0 is set to 1, and the first envelope is adjusted as follows:

 (1655)

and

 (1656)

Otherwise, there is no adjustment.

 (1657)

– If the coder types of current frame or previous frame is UNVOICED, the first envelope is adjusted as follows:

 (1658)

– If the coder types of current frame is not AUDIO, that is, , the first envelope is adjusted as follows:

 (1659)

– If the last core bit-rate is larger than 8000, and the first spectral envelope is larger than the average BWE signal envelope of the previous frame, that is, , the adjustment is by:

 (1660)

Otherwise, there is no adjustment:

 (1661)

– If the extended layer of the previous frame is different from the one of current frame, the adjustment is as follows:

 (1662)

Otherwise, there is no adjustment:

 (1663)

Finally, the spectral envelopes are adjusted as follows:

 (1664)

and

 (1665)

 (1666)

The spectral envelopesare used for the following frequency adjustment.

6.1.5.2.2.4 Frequency excitation generation

The base frequency excitation signalis generated from the MDCT coefficients  of the core decoded signal or from random noise depending upon the bit-rate and coder type . The core type flagis introduced. It is initialized to 1, and is set to 0 if the coder type is not AUDIO and the total bit-rate is not larger than 8000.

– The LF MDCT coefficients are obtained from the MDCT coefficients of core decoded signal:

 (1667)

– If the coder type of current frame is UNVOICED, the base frequency excitation signal is generated from linear congruential uniform random noise generator as follows:

 (1668)

where

 (1669)

Parameter  is initialized as 21211 and updated for each MDCT coefficient. It is noted that  is calculated for every frame.

– Otherwise, the base frequency excitation signal is copied as defined in subclause 5.2.6.2.1.5.

6.1.5.2.2.5 Frequency excitation normalization and spectral envelope de-normalization

In order to normalize the base frequency excitation to remove the original low frequency envelope information, the parameter of adaptive normalization length  is calculated depending on the HB signal class and the MDCT coefficients of core decoded signal:

– The 256 MDCT coefficients in the 0-6400 Hz frequency range,  are split into 16 sharpness bands (16 coefficients per band). In sharpness band *j*, if  and, the counter is incremented by one.

where , and the maximum magnitude of the spectral coefficients in a sharpness band, denoted , is:

 (1670)

Parameter  is initialized to 0 and calculated for every frame.

– Then the normalization length  is obtained:

 (1671)

where the current normalization length  is calculated depending on the HB signal class:

 (1672)

and the current normalization length is preserved as.

Then, according to the adaptive normalization length, the noise content of the base frequency excitation signal is generated, and the base frequency excitation signal is normalized to remove the core envelope information.

* The normalized envelope is calculated:

 (1673)

– If the bitrate is 7200 or 8000 and the coder type of current frame is not UNVOICED, the signs and amplitudes of HB coefficients are calculated by:

 (1674)

 (1675)

The adjusted coefficients are obtained by the amplitudes, the normalization envelopes and adaptive normalization length:

 (1676)

If , the adjusted coefficients are modified further by the modification factor , and then the base frequency excitation signal with the noise content is obtained by the signs and the adjusted coefficients:

 (1677)

where, the modification factor  for HARMONIC frame, otherwise, .

– Otherwise, there is no adjustment, that is,

 (1678)

– Next, the adjusted frequency excitation signal is normalized to remove the core envelope information:

 (1679)

– If the coder type of current frame is not UNVOICED, the frequency signal is adaptively adjusted further as follows:

 (1680)

where,  and , are defined in table 163.

Otherwise, there is no adjustment.

 (1681)

Table 163: Sub-band boundaries and number of coefficients per sub-band in NORMAL frames

|  |  |  |
| --- | --- | --- |
| *j* |  |  |
| 0 | 240 | 16 |
| 1 | 256 | 24 |
| 2 | 280 | 16 |
| 3 | 304 | 24 |
| 4 | 320 | - |

Finally, the spectral envelope is applied to the normalized excitation signal to obtain the HB coefficients.

Three parameters: the energy, two control factors,, are calculated according to the core type flag, coder type and bit-rate as follows:

 (1682)

 (1683)

 (1684)

When, if  and , or the coder type is GENERIC, the envelope adjustment flag  is set to 1 and the spectral envelope is adjusted:

 (1685)

where, is the energy of previous frame, and is the previous envelope adjustment flag. It is noted that the envelope adjustment flag is initialized to 0 and preserved for the next frame.

Otherwise, there is no adjustment:

 (1686)

The spectral envelopes are smoothed by the one between the current frame and the previous frame according to the following conditions, and when the current and previous frames apply different extended layer or the current frame is lost frame, the previous spectral envelopes are set to the current spectral envelopes.

When the current frame is NORMAL, or the current frame is HARMONIC and , the adjustment is as follows.

– If and, and at least one of the coder types of current and previous frames is AUDIO, that is, or , the smoothing process is performed by:

 (1687)

Else if the conditions: , , , ,,and , are all satisfied, the smoothing process is performed by:

 (1688)

where, is the energy of previous frame, and the is preserved for the next frame at the end of spectral envelope adjustment.

Else if the conditions: , , and , are all satisfied, the smoothing process is performed by:

 (1689)

Otherwise, there is no adjustment.

 (1690)

– The spectral envelope is further adjusted by:

 (1691)

where,  is the attenuation factor. It is initialized to 1 and set to  if  and are satisfied. are defined in table 164.

Table 164: The envelope attenuation factor

|  |  |
| --- | --- |
| *j* |  |
| 0 | 0.8000 |
| 1 | 0.7746 |
| 2 | 0.7483 |

– Then the adjusted spectral envelopes are applied to the excitation signal by:

 (1692)

When the conditions: the current frame is NORMAL frame, or the current frame is HARMONIC frame and , are not satisfied, the adjustment is as follows.

– The first spectral envelope is firstly processed as follows.

 (1693)

– If the conditions: , , are satisfied, the adjustment is performed by:

 (1694)

Otherwise, there is no adjustment:

 (1695)

– Then the adjusted spectral envelopes are applied to the excitation signal by:

 (1696)

where,  is the attenuation factor. It is initialized to 1 and set to  if  and .are defined in table 164.

The MDCT coefficients of HB signal are refined by:

 (1697)

And the spectral envelopes of current frame are preserved as for the next frame.

6.1.5.2.2.6 Windowing and frequency-to-time transformation

A 320-point inverse MDCT is used for WB FD BWE. Refer to subclause 6.2.4.

#### 6.1.5.3 Decoding of upper band at 64 kb/s

The upper band at 64 kbps bit-rate decoding starts with dequantizing the received spectrum coefficients by means of the AVQ as described in subclause 6.1.1.2.1.6.

The spectrum between 14.4 kHz and 16 kHz in SWB, resp. 20 kHz in FB, is reconstructed using decoded part of the upper band spectrum.

Further the spectral envelope is decoded and the quantized spectral envelope (four bands in normal mode or two bands in transient mode) is used to denormalize per envelope the decoded spectrum. Each spectral coefficient in an overlap region (7.6 kHz – 8 kHz) is multiplied by a factor lower than 1.0. The overlap region corresponds to the part of the spectrum where the ACELP lower band synthesis is suppressed due to the attenuation of the resampling filters. Consequently spectral coefficients in an overlap region equalize the spectral gap that would be present if the upper band coding would start at 8 kHz only.

Finally the spectrum is de-normalized by the decoded global gain and transformed to the time domain using iDCT and OLA function.

##### 6.1.5.3.1 Decoding in normal mode

The global gain and the spectral envelope corresponding to 4 sub-bands are decoded. The global gain  is de-quantized using a 5-bit log gain de-quantizer at the range of [3.0; 500.0]. The spectral envelopes  are de-quantized using two-dimensional VQs by means of 6bits and 5bits respectively as defined in subclause 5.2.6.3.1.

Then the band index with the minimum envelope  is calculated by, and the envelope of the spectrum between 14.4 kHz and 16 kHz in SWB, resp. 20 kHz in FB is predicted as follows:

1. If, .
2. If , the index of attenuation factor is decoded. Then, the is adjusted depending upon the index:

 ()

The number of the sub-bandsis obtained according to the number of the total bits  and the saturated threshold as follows:

 ()

The first stage sub-vectors  are decoded by means of AVQ and the spectrum of the upper band is obtained by the first stage sub-vectors ,

 ()

and the is saved to , i.e. .

If the number of the remaining bits after the first stage decoding is larger than 14 and the first stage quantized spectrum is non-zero, then the second stage decoding is performed. The second stage global gain, is decoded and updated by:

 ()

The number of the sub-bandsis obtained according to the remaining bitsand the saturated thresholdas follows:

 ()

The second stage sub-vectors  are also decoded by means of AVQ.

Then, the spectrum of the upper band is reconstructed by the contribution of the second stage decoding as follows:

– The counter  is initialized to 0.

– In the sub-band , , if the first stage AVQ codebook index , the spectrum of the upper band is adjusted by , and the second stage AVQ codebook index is added to the first stage AVQ codebook index . Then, the counter are incremented by 1.

– The sub-band indexis initialized to 0, while, if , the spectrum is adjusted by , and . Then, the counter is incremented by 1.

The spectrum is then reordered according to the number of the total bits.

– If , the reordered spectrum is obtained by:

 ()

where is the start frequency bin of spectrum reconstruction, and for normal frames.

– Otherwise, the reordered spectrum is obtained by:

 ()

where the and are set as follows:

 ()

 ()

And then the spectrum of the band with the minimum envelope is reconstructed as follows:

 ()

Meanwhile, the AVQ codebook index is adjusted as follows:

If ,

 ()

Otherwise,

 ()

And then,

 ()

Then the noise filling is applied to the spectrum. If the remaining bits after the above decoding, the 272 MDCT coefficients of the upper band  are divided into 34 sub-bands (8 coefficients per band). Two indices,  are introduced to select a base frequency band which is used to reconstruct the un-decoded coefficients in the sub-bands.

– The sub-band index is initialized to 0.

– If, in the index range from 0 to 34, the index of the first sub-band which AVQ codebook indexis searched , and then in the index range from the index to 34, the index  of the first sub-band which AVQ codebook indexis searched, and the sub-band index is set by,and. If , the  is adjusted by  , and then the MDCT coefficients in the sub-bands  are reconstructed by:

 ()

where, and  are the weighted factors, and , is obtained by:

 ()

 ()

the index of sub-band is from  to 0, and is initialized to  and updated by:

 ()

1. If , the index is incremented by 1; Otherwise, a base frequency band is selected to reconstruct the un-decoded coefficients as follows:

2. Initialize the indicesand , and then in the sub-band ,, search the index  of the first sub-band which AVQ codebook index, and set. If , the index and the position are further adjusted: , .

3. Then the MDCT coefficients of the upper band are adjusted by:

 ()

where is from to , and the  is initialized to  and it is adjusted by:

 ()

and,

 ()

Parameter  is initialized as 12345 and updated for each MDCT coefficient. It should be noted that  is calculated for every frame.

4. Finally, judge whether or not. If , return to step 1.

If the spectral tilt of the decoded core signal is larger than 5, the MDCT coefficients of the upper band are further adjusted. In sub-band, if, the adjustment is performed as follows:

 ()

where is obtained by the algorithm described in equation (629) and (630), andis obtained as follows:

 ()

and if ,  is refined by:

 ()

When , the coefficients of the upper band in the index range [504, 511] are smoothed by:

 ()

where  is defined as follows:

 ()

The spectrum between 14.4 kHz and 16 kHz in SWB, resp. 20 kHz in FB is reconstructed by:

 ()

where is the number of the coefficients between 14.4 kHz and 16 kHz in SWB, resp. 20 kHz in FB:

 ()

Then  or the coefficients in the index range [576, 583] are smoothed by,

 ()

where  is defined as follows:

 ()

and the spectrum is de-normalized using the spectral envelope by:

 ()

Otherwise the spectrum is obtained as:

 (1727a)

where  for 32 kHz sampled output or  for 48 kHz smapled output. Next the overlap coefficients , which depend on the output sampling rate (32 kHz or 48 kHz), defined in table165 are used for adjustment as follows:

 (1728)

Table 165: Overlap coefficients 

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| *k* | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| , 32 kHz | 0.27 | 0.306 | 0.324 | 0.351 | 0.378 | 0.396 | 0.414 | 0.4275 |
| , 48 kHz | 0.30 | 0.34 | 0.36 | 0.39 | 0.42 | 0.44 | 0.46 | 0.475 |
| *k* | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
| , 32 kHz | 0.441 | 0.459 | 0.486 | 0.513 | 0.558 | 0.648 | 0.747 | 0.855 |
| , 48 kHz | 0.49 | 0.51 | 0.54 | 0.57 | 0.62 | 0.72 | 0.83 | 0.95 |

Finally the decoded global gain  is applied to adjust the spectrum of the upper band.

 ()

The MDCT coefficients are preserved for the next frame as follows:

 ()

Two parameters of and  are updated by:

 ()

 ()

##### 6.1.5.3.2 Decoding in transient mode

There are 4 sub-fames for transient mode. In sub-frame,, the global gain and spectral envelopes , are decoded. The global gain of each sub-frame is de-quantized using a 5-bit log gain de-quantizer at the range of [3.0; 500.0]. The first sub-frame spectral envelopes  are de-quantized firstly using two-dimensional VQs by means of 4 bits codebook defined in subclause 5.2.6.3.2. The indices of the first sub-frame spectral envelopes is noted as .

If , which means the first sub-frame spectral envelopes are de-quantized in the first part of the 4 bits codebook, the spectral envelope  of the following three sub-frame are de-quantized using two-dimensional VQs by means of 3 bits codebook , i.e. ,  as described in subclause 5.2.6.3.2.

If , the spectral envelope  of the following three sub-frame are de-quantized using two-dimensional VQs by means of 3 bits codebook, i.e. ,  as described in subclause 5.2.6.3.2.

And then the spectrum of the upper band is reconstructed.

The number of the coefficients between 14.4 kHz and 16 kHz in SWB, resp. 20 kHz in FB,  , is set by:

 ()

And the envelope of the MDCT coefficients between 14.4 kHz and 16 kHz in SWB, resp. 20 kHz in FB is calculated by:

1. If 

 ()

1. If , the index of attention factor is decoded, and then the is adjusted depending upon the index.

 ()

The normalized spectrum of the upper band is decoded by the sub-vectors which are obtained by means of the AVQ:

 ()

where is the start frequency bin of spectrum reconstruction, and for transient frames. is the frame length. for SWB signal and for FB signal.

Then the noise filling is applied to the spectrum. The first 64 MDCT coefficients of the upper band  are divided into 8 sub-bands (8 coefficients per band). In the sub-band , , if the AVQ codebook index is equal to 0, that is,, the signal is generated from linear congruential uniform random noise generator as follows:

 ()

where

 ()

Parameter  is initialized as 12345 and updated for each MDCT coefficient. It should be noted that  is calculated for every frame.

If the spectral tilt of the decoded core signal is larger than 5, the MDCT coefficients of the upper band are further adjusted. If, the adjustment is by:

 ()

where is obtained by the algorithm described in equations (629) and (630), andis obtained as follows:

 ()

and if ,  is refined by:

 ()

The spectrum between 14.4 kHz and 16 kHz in SWB, resp. 20 kHz in FB is reconstructed by replicating the nearby coefficients:

 ()

Then the spectrum of the upper band is de-normalized using the spectral envelope by:

 ()

and the modification factors defined in table 165 are used to adjust the overlapped coefficients as follows:

 ()

where, .

Finally, the decoded global gain  is applied to adjust the spectrum of the upper band.

 ()

##### 6.1.5.3.3 Windowing and frequency-to-time transformation

A 640-point (SWB) or 960-point (FB) inverse MDCT is used for the upper band frequency signal. Refer to subclause 6.2.4.

##### 6.1.5.3.4 Post-processing in temporal domain

If the current frame is a good one, further adjustment of the upper band synthesized temporal signal is needed. The synthesized temporal signal of the upper band is divided into 4 sub-frames, and the energy of each sub-frame , is computed by:

 ()

For each sub-frame, the long term energy  is initialized to 0, and updated according to the following equation:

 ()

In the above equation, the weighted factor  is set to 0.25, and the convention is that for the first sub-frame,

 from the previous frame. For each sub-frame , a comparison between the short term energy  and the long term energy is performed. A transient is detected whenever the energy ratio is above a certain threshold and the sub-frame indexis recorded as. Formally, a transient is detected whenever:

 ()

where  is the energy ratio threshold and is set to .

When the current frame is transient and the conditions:, ,, are all satisfied, the adjustment is processed.

 is obtained depending upon whether the current and previous frames apply the same extend layer:

 ()

where the energyis calculated by:

 ()

 ()

then,

 ()

The signal class  of the current frame is preserved as  for the next frame.

When or , if , the post-processing is performed in case of switching of different extend layers or different cores. The gain factor is first calculated by:

 ()

where is the index of the time sample with the maximum magnitude, and

 ()

The gain factor  is refined by:

 ()

where and are obtained by:

 ()

 ()

Then the synthesized signal of the upper band is adjusted by:

 ()

where is calculated as follows:

 ()

 ()

And the preserved synthesized signal of the upper band for OLA function, , is adjusted by:

 ()