# 5 Functional description of the encoder

The description of the encoder is as follows. First, common pre-processing is described, and then the different elements of the encoder are described one by one. The discontinuous transmission (DTX) operation and the AMR-WB interoperable option are then given in separate subclauses, again referencing the same processing as in the default option.

## 5.1 Common processing

### 5.1.1 High-pass Filtering

The input audio signal sampled at 8, 16, 32 or 48 kHz,  being the sample index, is high-pass filtered to suppress undesired low frequency components. The transfer function of the HP filter has a cut-off frequency of 20 Hz (–3 dB) and is given by

 (1)

The coefficients of the HP filter for a given input sampling frequency are given in the table below.

Table : Coefficients of the 20Hz HP filer

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 8 kHz | 16 kHz | 32 kHz | 48 kHz |
|  | 0.988954248067140 | 0.994461788958195 | 0.997227049904470 | 0.998150511190452 |
|  | -1.977908496134280 | -1.988923577916390 | -1.994454099808940 | -1.996301022380904 |
|  | 0.988954248067140 | 0.994461788958195 | 0.997227049904470 | 0.998150511190452 |
|  | 1.977786483776764 | 1.988892905899653 | 1.994446410541927 | 1.996297601769122 |
|  | -0.978030508491796 | -0.988954249933127 | -0.994461789075954 | -0.996304442992686 |

The input signal, filtered by the HP filter, is denoted as .

### 5.1.2 Complex low-delay filter bank analysis

#### 5.1.2.1 Sub-band analysis

The audio signal  is decomposed into complex valued sub-bands by a complex modulated low delay filter bank (CLDFB). Depending on the input sampling rate , the CLDFB generates a time-frequency matrix of 16 time slots and  sub-bands where the width of each sub-band is 400 Hz.

The analysis prototype  is an asymmetric low-pass filter with an adaptive length depending on. The length of  is given by  meaning that the filter spans over 10 consecutive blocks for the transformation. The prototype of the LP filter has been generated for 48 kHz. For other input sampling rates, the prototype is obtained by means of interpolation so that an equivalent frequency response is achieved. Energy differences in the sub-band domain caused by different transformation lengths are compensated for by an appropriate normalization factors in the filter bank. The following figure shows the plot of the LP filter prototype  for of 48 kHz.

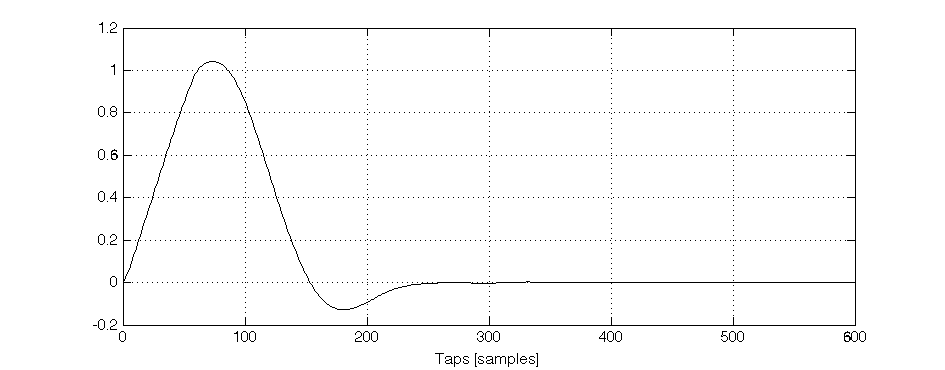


Figure 4 : Impulse response of CLDFB prototype filter with 600 taps for 48 kHz sample rate

The filter bank operation is described in a general form by the following formula:

 (2)

where  and  are the real and the imaginary sub-band values, respectively, is the sub-band time index with , index is defined as , is the modulation offset of  andis the sub-band index with .

As the equations show, the filter bank is comparable to a complex MDCT but with a longer overlap towards the past samples. This allows for an optimized implementation of CLDFB by adopting DCT-IV and DST-IV frameworks.

#### 5.1.2.2 Sub-band energy estimation

The energy in the CLDFB domain is determined for each time index  and frequency sub-band by

 (3)

Furthermore, energy per-band is calculated by summing up the energy values in all time slots. That is

 ()

In case , additional high frequency energy value  is calculated for the frequency range from 8kHz to 16kHz by summing up  over one frame which is delayed by one time slot.

 ()

 is further scaled to an appropriate energy domain. In case the high bands are not active, is initialized to the maximum value.

### 5.1.3 Sample rate conversion to 12.8 kHz

The linear predictive (LP) analysis, the long-term prediction (LTP), the VAD algorithm and signal are performed at the 12.8 kHz sampling rate. The HP-filtered input signal  is therefore converted from the input sampling frequency to 12.8 kHz.

#### 5.1.3.1 Conversion of 16, 32 and 48 kHz signals to 12.8 kHz

For 16, 32 and 48 kHz signals, the sampling conversion is performed by first up‑sampling the signal to 192 kHz, then filtering the output through a low-pass FIR filter  that has the cut‑off frequency at 6.4 kHz. Then, the signal is down-sampled to 12.8 kHz. The filtering delay is 15 samples at 16 kHz sampling frequency which corresponds to 0.9375 ms.

The up-sampling is performed by inserting 11, 5 or 3 (for 16, 32 or 48 kHz, respectively) zero-valued samples between each 2 samples for each 20-ms frame of 320 samples (at 16 kHz sampling frequency)

 (6)

where  is the signal at 192 kHz sampling frequency and  is the up-sampling factor equal to 12 for a 16 kHz input, 6 for a 32 kHz input and 4 for a 48 kHz input. Then, the signal  is filtered through the LP filter  and decimated by 15 by keeping one out of 15 samples. The filter  is a 361-tap linear phase FIR filter having a cut-off frequency of 6.4 kHz in the 192 kHz up-sampled domain. The filtering and decimation can be done using the relation

 (7)

where  is the impulse response of . The operations in equations (6) and (7) can be implemented in one step by using only a part of the filter coefficients at a time with an initial phase related to the sampling instant *n*. That is

 (8)

In case the encoder is externally forced to narrow-band processing of the input signal, the cut-off frequency of the LP filter is changed from 6.4 kHz to 4 kHz.

#### 5.1.3.2 Conversion of 8 kHz signals to 12.8 kHz

For 8 to 12.8 kHz resampling a sharper resampling filter is beneficial. Double length low-pass FIR filter  is used in this case. The doubling of the impulse response length is compensated by a low delay resampling method. The filter  is a 241-tap linear phase FIR filter having a cut-off frequency of 3.9 kHz and is applied in the up-sampled domain which is 64 kHz. Direct FIR filtering with this filter would yield a delay of 120/64 = 1.875 ms. In order to reduce this delay to 0.9375 ms, future samples are determined at 8 kHz by adaptive linear prediction. The exact number of future samples is found based on the difference between the actual delay (1.875 ms) and the desired delay (0.9375 ms) at 8 kHz. Therefore  future samples are predicted. These predicted samples are concatenated at the end of the current frame to form a support vector. Then, the sample rate conversion of  is performed in a similar way as for the other sampling rates, i.e.  is first up-sampled to 64 kHz, the output is filtered through the low-pass FIR filter  and the resulting signal is down-sampled to 12.8 kHz. The final filtering delay is aligned with that of the other resampling configurations, i.e 12 samples at 12.8 kHz sampling frequency which corresponds to 0.9375 ms.

To determine the future samples, linear prediction coefficients of order 16 are computed in the pre-emphasized domain in the following way. The last *L*ss =120 samples of the input frame  at 8 kHz are windowed by an asymmetrical analysis window *win*ss\_120:

 ()

and a first order autocorrelation analysis is made on the windowed signal . The pre-emphasis coefficient ss is obtained by

 ()

where *rw*(0) and *rw*(1) are the autocorrelation coefficients

 ()

The last 120 samples of the signal  are pre-emphasized using the adaptive filter

 ()

to obtain the pre-emphasized signal  of *L*ss =120 samples. Then  is windowed by the asymmetrical analysis window *win*ss\_120 and a 16th order autocorrelation analysis is made on the windowed signal 

 ()

These autocorrelation coefficients are lag-windowed by

 ()

where *wlag8k*(*k*) is defined as

 ()

Based on the autocorrelation coefficients *rpwl*(*k*), the linear prediction coefficients *ass*(*k*) are computed by the Levinson-Durbin algorithm. The future samples in the pre-emphasized domain  are predicted by zero input filtering through the 1/*Ass*(*z*) synthesis filter

 ()

Finally, the concatenated signal is de-emphasized through the filter . Note that only the last 7 predicted samples need to be de-emphasized. These 7 de-emphasized samples are concatenated to  (at positions *n* = 160,…,166) to form the support vector.

The up-sampling of  is then performed by inserting 7 zero-valued samples between each 2 samples for each 20-ms frame of 160 samples (at 8 kHz sampling frequency) completed by 7 predicted future samples (167 in total)

 (17)

where  is the signal at 64 kHz sampling frequency. Then, the signal  is filtered through the LP filter  and decimated by 5 by keeping one out of 5 samples. The filtering and decimation can be done using the relation

 (18)

where  is the impulse response of  and  assures that the index of *s*64 is never higher than the highest available index for (which is 1335). Indeed, it corresponds to the delay of this filtering at 64 kHz. To reduce complexity, the operations in equations (17) and (18) can be implemented in one step by using only a part of the filter coefficients at a time with an initial phase related to the sampling instant *n*. This polyphase implementation of the resampling filter is applied on the concatenated support vector. That is

 ()

where  is derived from the delay of this filtering at 8 kHz. It assures that the index of *sHPC* is never higher than the highest available index (which is 166).

#### 5.1.3.3 Conversion of input signals to 16, 25.6 and 32 kHz

If ACELP core is selected for WB, SWB or FB signals at bitrates higher than 13.2 kbps (see subclause 5.1.16), its internal sampling rate is set to 16 kHz rather than 12.8 kHz. If the input signal is sampled at 8 kHz, there is no conversion needed because for NB signals, ACELP core is always operated at 12.8 kHz. If the input signal is sampled at 16 kHz, no conversion is needed either and the input signal is only delayed by 15 samples which corresponds to 0.9375 ms. This is to keep all pre-processed signals aligned regardless of the bitrate or bandwidth. Thus, the input signal is resampled to 16 kHz only if its sampling frequency is 32 or 48 kHz.

The resampling operation is done in the same way as for the case of 12.8 kHz (see subclause 5.1.3.1), i.e. by means of FIR filtering. The coefficients of the LP filter are different but the filtering delay is still the same, i.e. 0.9375 ms.

The resampled signal is denoted  where *n*=0,..,319.

The input signal is converted to 25.6 kHz at 48 kbps and to 32 kHz at 96 or 128kbps but only for SWB and FB signals. The sampling conversion is again similar as in the case of 12.8 kHz with differences in LP filter coefficients. The resampled signals are denoted  and , respectively.

### 5.1.4 Pre-emphasis

A first-order high-pass filter is used to emphasize higher frequencies of the input signal and it is given by

 ()

where  is the pre-emphasis factor which is set to 0.68. The input signal to the pre-emphasis filter is  and the output signal is denoted .

If ACELP core is selected for WB, SWB or FB signals at bitrates higher than 13.2 kbps (see subclause 5.1.16), its internal sampling rate is 16kHz rather than 12.8kHz. In this case, the pre-emphasis is re-done at the end of the pre-processing chain, on  with . The resulting signal is denoted .

If MDCT-based TCX is selected for SWB or FB high-rate LPC configurations, is used as pre-emphasis factor when pre-emphasis is applied to signals at a sampling rate higher than 16kHz.

### 5.1.5 Spectral analysis

Spectral analysis is used in the encoder for signal activity detection (SAD) and signal classification functions. The discrete Fourier transform (DFT) is used to perform the spectral analysis and spectral energy estimation.

#### 5.1.5.1 Windowing and DFT

The frequency analysis is done twice per frame using 256-point fast Fourier transform (FFT) with a 50% overlap. The centre of the first window is placed 96 samples past the beginning of the current frame. The centre of the second window is placed 128 samples farther, i.e., in the middle of the second subframe of the current frame. A square root of a Hanning window (which is equivalent to a sine window) is used to weight the input signal for the frequency analysis. The square root Hanning window is given by

 ()

where = 256 is the size of FFT analysis. Note that only half of the window is computed and stored since it is symmetric (from 0 to ).



Figure 5: Relative positions of the spectral analysis windows

The windowed signal for both spectral analyses is obtained as:

 ()

whereis the pre-emphasized input signal (is the first sample in the current frame). The superscripts [0] and [1] used to denote the first and the second frequency analysis, respectively, are dropped for simplicity. An FFT is performed on both windowed signals to obtain two sets of spectral parameters per frame:

 ()

The output of the FFT provides the real and the imaginary parts of the spectrum denoted as,  and , . Note, that  corresponds to the spectrum at 0 Hz (DC) and  corresponds to the spectrum at 6400 Hz. The spectrum at these points is only real-valued and usually ignored in the subsequent analysis.

After the FFT analysis, the resulting spectrum is divided into critical bands [17] using the intervals having the following limits (20 bands in the frequency range 0-6400 Hz):

Table : Critical bands

|  |  |  |  |
| --- | --- | --- | --- |
| band |  |  |  |
| 0 | 0 | 100 | 2 |
| 1 | 100 | 200 | 2 |
| 2 | 200 | 300 | 2 |
| 3 | 300 | 400 | 2 |
| 4 | 400 | 510 | 2 |
| 5 | 510 | 630 | 2 |
| 7 | 630 | 770 | 3 |
| 6 | 770 | 920 | 3 |
| 8 | 920 | 1080 | 3 |
| 9 | 1080 | 1270 | 4 |
| 10 | 1270 | 1480 | 4 |
| 11 | 1480 | 1720 | 5 |
| 12 | 1720 | 2000 | 6 |
| 13 | 2000 | 2320 | 6 |
| 14 | 2320 | 2700 | 8 |
| 15 | 2700 | 3150 | 9 |
| 16 | 3150 | 3700 | 11 |
| 17 | 3700 | 4400 | 14 |
| 18 | 4400 | 5300 | 18 |
| 19 | 5300 | 6350 | 21 |

The 256-point FFT results in a frequency resolution of 50 Hz (i.e., 6400/128 Hz). Thus, after ignoring the DC component of the spectrum, the number of frequency bins per critical band are given in the last column, denoted .

#### 5.1.5.2 Energy calculations

The spectral analysis module also calculates several energy-related parameters. For example, an average energy per critical band is computed as

 (24)

whereandare, respectively, the real and the imaginary parts of the-th frequency bin and  is the index of the first bin in the *i*th critical band given by={1, 3, 5, 7, 9, 11, 13, 16, 19, 22, 26, 30, 35, 41, 47, 55, 64, 75, 89, 107}. Furthermore, energy per frequency bin,, is calculated as

 (25)

Finally, the spectral analysis module computes the average total energy for both FFT analyses in a 20 ms frame by summing the average critical band energies. That is, the spectrum energy for the first spectral analysis window is computed as

 ()

and, similarly, the second frame energy, denoted as.

The total frame energy (in dB) is computed as the average of the two frame energies. That is

 (27)

The total energy per frequency bin (power spectrum) is calculated as

 (28)

The output parameters of the spectral analysis module (both spectral analyses), that is the average energy per critical band, the energy per frequency bin and the total energy in dB, are used in several subsequent functions.

Note that, for narrow band inputs sampled at 8 kHz, after sampling conversion to 12.8 kHz, there is no content at both ends of the spectrum. Thus, the lowest critical band as well as the last three critical bands are not considered in the computation of output parameters (only bands from are considered).

In addition to the absolute frame energy, calculated in (27), relative energy of the frame is calculated as the difference between the total frame energy in dB and the long-term active signal energy. The relative frame energy is given by

 (29)

The long-term active signal energy is updated only during active frames (explained in subclause 5.1.12.5). Note that the long-term active signal energyis updated only after the signal activity detection module.

### 5.1.6 Bandwidth detection

A detection algorithm is applied to detect the actual input audio bandwidth for input sampling rates greater than 8 kHz. This bandwidth information is used to run the codec in its optimal mode, tailored for a particular bandwidth (BW) rather than for a particular input sampling frequency. For example, if the input sampling frequency is 32 kHz but there is no "energetically" meaningful spectral content above 8 kHz, the codec is operated in the WB mode. The following bandwidths/modes are used throughout the EVS codec: NB (0-4kHz), WB (0-8kHz), SWB (0-16kHz) and FB (0-20 kHz).

The detection algorithm is based on computing energies in spectral regions and comparing them to certain thresholds. The bandwidth detector operates on the CLDFB values (see subclause 5.1.2). In the AMR-WB IO mode, the bandwidth detector uses a DCT transform to determine the signal bandwidth.

#### 5.1.6.1 Mean and maximum energy values per band

The CLDFB energy vector computed per 400Hz frequency bins (see subclause 5.1.2.2), is further aggregated as described below. Each value of  represents a 1600Hz band consisting of four CLDFB energy bins summed up from  to .

 ()

Depending on the input sampling frequency up to nine CLDFB bands are calculated usingthe above equation and the values are given below:

Table : CLDFB bands for energy calculation

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  | bandwidth in kHz | bandwidth index |
| 0 | 3 | 6 | 1.2 – 2.8 | NB |
| 1 | 11 | 14 | 4.4 – 7.2 | WB |
| 2 | 14 | 17 |
| 3 | 23 | 26 | 9.2 – 15.6 | SWB |
| 4 | 27 | 30 |
| 5 | 31 | 34 |
| 6 | 35 | 38 |
| 7 | 42 | 45 | 16.8 – 20.0 | FB |
| 8 | 46 | 49 |

The values in CLDFB bands are converted to the log domain and scaled by

 ()

where  is set according to the input sampling frequency as follows: 88.293854 for 8kHz, 88.300926 for 16kHz, 88.304118 for 32kHz and 88.028412 for 48kHz.

The per-band CLDFB energy is then used to calculate the mean energy values per bandwidth:

 ()

and the maximum energy values per bandwidth:

 ()

In case of the DCT based detector, the DCT values are computed by first applying a Hanning window on the 320 samples of the input audio signal sampled at input sampling rate. Then the windowed signal is transformed to the DCT domain and finally decomposed into several bands as shown in Table 3a.

Table 3a: DCT bands for energy calculation

|  |  |  |
| --- | --- | --- |
|  | bandwidth in kHz | bandwidth index |
| 0 | 1.5 – 3.0 | NB |
| 1 | 4.5 – 7.5 | WB |
| 2 |
| 3 | 9.0 – 15.0 | SWB |
| 4 |
| 5 |
| 6 |
| 7 | 16.5 – 19.5 | FB |
| 8 |

The values in DCT bands are converted to the log domain by

 (33a)

and per-band and maximum energies are computed using (32) and (33) while the constant 1.6 in these equations is omitted in case of the DCT based detector.

### 5.1.7 Bandwidth decision

The following decision logic is identical for CLDFB and DCT versions of energy calculations, except for some constants which were adapted to get similar detection results.

The long-term CLDFB energy mean values for NB, WB and SWB are updated as follows:

 ()

where the superscript [-1] has been used to denote the value of  in the previous frame. The update takes place only if the local SAD decision is active and only if the long-term background noise level, , is higher than 30 dB.

The values are compared to certain thresholds also taking the current maximum values into account, which leads to increasing or decreasing counters for each bandwidth as described below in the flowchart.



Figure 6: Increasing and decreasing of BW counters

The tests in the above diagram are performed sequentially from top to bottom. The BW counters are then used to decide the actual signal bandwidth, *BW*, according to the logic described in the following schematic diagram.



Figure 7: BW selection logic

In the above diagram, the tests are performed in a sequential order, i.e. it could happen that decision about signal bandwidth is changed several times in this logic. After every selection of a particular bandwidth, certain counters are reset to their minimal value 0 or to their maximum value 100.

Finally, the resulting bandwidth can be upper limited in case the codec performance has not been optimized for it at particular bitrate. For example, at 9.6 kbps, the codec supports coding up to SWB. Therefore, if the detected bandwidth is FB, it is overwritten to SWB at this bitrate. The following table shows the range of bitrates for which the codec performance has been optimized for each bandwidth.

Table : Optimization of the codec performance per bandwidth

|  |  |
| --- | --- |
| bandwidth | bitrate range [kbps] |
| NB | 7.2 - 24.4 |
| WB | 7.2 - 128 |
| SWB | 9.6 - 128 |
| FB | 16.4 - 128 |

### 5.1.8 Time-domain transient detection

The HP-filtered input signal  including the look-ahead is input to the time-domain transient detector. The HP-filtered input signal  is further high-pass filtered. The transfer function of the transient detection’s HP filter is given by

 ()

The signal, filtered by the transient detection’s HP filter, is denoted as . The HP-filtered signal  is segmented into 8 consecutive segments of the same length. The energy of the HP-filtered signal  for each segment is calculated as:

 ()

where  is the number of samples in 2.5 milliseconds segment at the input sampling frequency. An accumulated energy is calculated using:

 ()

A transient is detected if the energy of a segment  exceeds the accumulated energy by a constant factor of 8.5and the attack index is set to . If no attack is detected but strong energy increase is detected in segment, the attack index is set to and the frame is not marked as a transient frame.

The energy change for each segment is calculated as:

 ()

The temporal flatness measure is calculated as:

 ()

The maximum energy change is calculated as:

 ()

If index of  or  is negative then it indicates a value from the previous segment, with segment indexing relative to the current frame. is the number of the segments from the past frames. It is equal to 0 if the temporal flatness measure is calculated for the usage in ACELP/TCX decision. If the temporal flatness measure is calculated for the TCX LTP decision then it is equal to:

 ()

 is the number of segments from the current frame. It is equal to 8 for non-transient frames. For transient frames first the locations of the segments with the maximum and the minimum energy are found:

 ()

If  then is set to , otherwise  is set to 8.

### 5.1.9 Linear prediction analysis

Short-term prediction or linear prediction (LP) analysis using the autocorrelation approach determines the coefficients of the synthesis filter of the CELP model. The autocorrelation of windowed speech is converted to the LP coefficients using the Levinson-Durbin algorithm. Then, the LP coefficients are transformed to the line spectral pairs (LSP) and consequently to line spectral frequencies (LSF) for quantization and interpolation purposes. The interpolated quantized and unquantized coefficients are converted back to the LP domain to construct the synthesis and weighting filters for each subframe.

#### 5.1.9.1 LP analysis window

In case of encoding of an active signal frame, two sets of LP coefficients are estimated in each frame using a 25 ms asymmetric analysis window (320 samples at 12.8 kHz sampling rate), one for the frame-end and one for mid-frame LP analysis. A look ahead of 8.75ms (112 samples at 12.8 kHz sampling rate) is used for the frame-end autocorrelation calculation. The frame structure is shown below.



Figure 8: Relative positions and length of the LP analysis windows

The frame is divided into four sub-frames, each having a length of 5 ms, i.e., 64 samples. The windows for frame-end analysis and for mid-frame analysis are centred at the 2nd and 4th sub-frame of the current frame, respectively. An asymmetrical window with the length of 320 samples is used for windowing. The windowed signal for mid-frame is calculated as

 ()

and the windowed signal for frame-end is calculated as

 (44)

#### 5.1.9.2 Autocorrelation computation

The autocorrelations of the windowed signal are computed by

 (45)

where  is set to 320. When ,  is set to 100 as well.

#### 5.1.9.3 Adaptive lag windowing

In addition, bandwidth expansion is applied by lag windowing the autocorrelations using the following window

 ()

where  is the sampling frequency (12800 or 16000) and the bandwidth frequency  is set adaptively based on the OL pitch lag  in the 12.8 kHz domain and the normalized correlation  (i.e., pitch gain). These parameters are obtained in the OL pitch estimation module from the look-ahead part of the current or the previous frame, depending on whether the adaptive lag windowing is applied before or after the OL pitch estimation. In some special cases,  and are used instead of  and , respectively. These situations will be described later in this specification. Note that the shorter pitch lag and/or the larger pitch gain, the stronger (heavy smoothing with larger) window is used to avoid excessive resonance in the frequency domain. The longer pitch lag and/or the smaller normalized correlation, the weaker window (light smoothing with smaller) is used to get more faithful representation of the spectral envelope.

Table 5: Selection of band width frequency  in Hz

|  |  |  |  |
| --- | --- | --- | --- |
|  | < 80 | 80<= < 160 | 160<= |
| 0.6 < | 60 | 40 | 20 |
| 0.3 <<=0.6 | 40 | 40 | 20 |
| <= 0.3 | 40 | 20 | 20 |

The modified autocorrelation function,  is calculated as

 ()

Further,  is multiplied by the white noise correction factor 1.0001 which is equivalent to adding a noise floor of -40 dB.

#### 5.1.9.4 Levinson-Durbin algorithm

The modified autocorrelation function,, is used to obtain the LP filter coefficients  by solving the set of equations:

 (48)

The set of equations in () is solved using the Levinson-Durbin algorithm. This algorithm uses the following recursion:

 ()

The final solution is given as. The residual error energies (LP error energies)  are also used in the subsequent processing.

#### 5.1.9.5 Conversion of LP coefficients to LSP parameters

The LP filter coefficients  are converted to the LSP representation [16] for quantization and interpolation purposes. For a 16th-order LP filter, the LSPs are defined as the roots of the sum and difference polynomials

 ()

The polynomials  and  are symmetric and asymmetric, respectively. It can be proved that all roots of these polynomials lie on the unit circle and are interlaced. The polynomials  and  have each 8 conjugate roots, denoted  and called the Line Spectral Pairs (LSPs). The corresponding angular frequencies  are the Line Spectral Frequencies (LSFs). The LSFs satisfy the ordering property . The coefficients of these polynomials are found by the following recursive relations:

 ()

where *M* = 16 is the predictor order.

The LSPs are found by evaluating the polynomials  and at 100 points equally spaced between 0 and π and checking for sign changes. A sign change indicates the existence of a root and the sign change interval is then divided four times to track the root precisely. Considering the conjugate symmetry of the polynomials  and and removing the linear term, it can be shown that the polynomials  and  can be written (considering) as

 ()

Considering the frequency mapping  we can define

 ()

an *m*th-order Chebyshev polynomial in *x* [18]. The polynomials  and  can then be rewritten using this Chebyshev polynomial expansion as

 ()

Neglecting the factor of 2, which does not affect the root searching mechanism, the series to be evaluated can be generalized to

 (55)

The Chebyshev polynomials satisfy the order recursion

 ()

with initial conditions . This recursion can be used to calculate  and . Then,  can be expressed in terms of  and 

 ()

The details about Chebyshev polynomial evaluation method can be found in [18].

In the following part of this document, the LSPs found by the described method will be denoted as , *i*=1,..,16 with .

#### 5.1.9.6 LSP interpolation

The LP parameters for each subframe are obtained by means of interpolation between the end-frame parameters of the current frame, the mid-frame parameters of the current frame and the end-frame parameters of the previous frame. However, the LP parameters are not particularly suitable for interpolation due to stability issues. For this reason, the interpolation is done on the respective LSP parameters and then converted back to the LP domain.

Let  denote the end-frame LSP vector of the current frame,  the mid-frame LSP vector of the current frame, both calculated by the method described in the previous section. Furthermore, let  be the end-frame LSP vector of the previous frame. The interpolated LSP vectors for all subframes are then given by

 ()

The same formula is used for interpolation of quantized LSPs described later in this document.

#### 5.1.9.7 Conversion of LSP parameters to LP coefficients

Once the interpolated LSP vectors are calculated, they are converted back into LP filter coefficients for each subframe. Each LSP parameter  gives rise to a second order polynomial factor of the form . These can be multiplied together to form the polynomials, i.e.

 ()

By using the Chebyshev polynomial expansion defined in (55) we can apply the following recursion to find the coefficients of the polynomials:

 ()

The coefficients  are computed similarly, by replacing  by , and with initial conditions and . Once the coefficients  and  are found, they are multiplied by  and , respectively, to form the polynomials  and . That is

 ()

Finally, the LP coefficients are found by

 ()

with . This is directly derived from the equation , and considering the fact that  and  are symmetric and asymmetric polynomials, respectively. The details of this procedure can be found in [18].

#### 5.1.9.8 LP analysis at 16kHz

If ACELP core is selected for WB, SWB or FB signals at bitrates higher than 13.2 kbps, its internal sampling rate is set to 16 kHz rather than 12.8 kHz. In this case, the LP analysis is done at the end of the pre-processing chain on input signal resampled to 16 kHz and pre-emphasized (see subclauses 5.1.3.3 and 5.1.4). In this case, the length of the LP analysis window is 400 samples at 16 kHz, which corresponds again to 25 ms. The windowed signal for mid-frame is calculated as

 ()

and the windowed signal for frame-end is calculated as

 ()

The autocorrelation computation, adaptive lag windowing and the conversion of LP coefficients to LSP parameters are performed similarly as in subclauses 5.1.9.2 thru 5.1.9.5. However, the LSP interpolation is done on 5 sub-frames instead of 4 sub-frames. The interpolated LSP vectors are given by



The conversion of LSP parameters to LP coefficients is then performed similarly as in subclause 5.1.9.7. At the end of the LP analysis there are *M*=16 LSP parameters and  coefficients but the corresponding LSFs span the range of 0-8000 Hz rather than 0-6400 Hz.

The LP analysis at 25.6 kHz and 32 kHz is described later in this document.

### 5.1.10 Open-loop pitch analysis

The Open-Loop (OL) pitch analysis calculates three estimates of the pitch lag for each frame. This is done in order to smooth the pitch evolution contour and to simplify the pitch analysis by confining the closed-loop pitch search to a small number of lags around the open-loop estimated lags.

The OL pitch estimation is based on a perceptually weighted pre-emphasized input signal. The open-loop pitch analysis is performed on a signal decimated by two, i.e. sampled at 6.4 kHz. This is in order to reduce the computational complexity of the searching process. The decimated signal is obtained by filtering the signalthrough a 5th-order FIR filter with coefficients {0.13, 0.23, 0.28, 0.23, 0.13} and then down-sampling the output by 2*.*

The OL pitch analysis is performed three times per frame to find three estimates of the pitch lag: two in the current frame and one in the look‑ahead area. The first two calculations are based on 10‑ms segments of the current frame. The final (third) estimation corresponds to the look-ahead, and the length of this segment is 8.75ms.

#### 5.1.10.1 Perceptual weighting

Perceptual weighting is performed by filtering the pre-emphasized input signal  through a perceptual weighting filter, derived from the LP filter coefficients. The traditional perceptual weighting filter  has inherent limitations in modelling the formant structure and the required spectral tilt concurrently. The spectral tilt is pronounced in speech signals due to the wide dynamic range between low and high frequencies. This problem is eliminated by introducing the pre-emphasis filter (see subclause 5.1.4) at the input which enhances the high frequency content. The LP filter coefficients are found by means of LP analysis on the pre-emphasized signal. Subsequently, they are used to form the perceptual weighting filter. Its transfer function is the same as the LP filter transfer function but with the denominator having coefficients equal to the de-emphasis filter (inverse of the pre-emphasis filter). In this way, the weighting in formant regions is decoupled from the spectral tilt. Finally, the pre-emphasized signal is filtered through the perceptual filter to obtain a perceptually weighted signal, which is used further in the OL pitch analysis.

The perceptual weighting filter has the following form

 ()

where

 ()

and  and . Since  is computed based on the pre-emphasized signal, the tilt of the filter  is less pronounced compared to the case when  is computed based on the original signal. The de-emphasis is also performed on the output signal in the decoder. It can be shown that the quantization error spectrum is shaped by a filter having a transfer function . Thus, the spectrum of the quantization error is shaped by a filter whose transfer function is , with  computed based on the pre-emphasized signal. The perceptual weighting is performed on a frame-by-frame basis while the LP filter coefficients are calculated on a sub-frame basis using the principle of LSP interpolation, described in subclause 5.1.9.6. For a sub-frame size  = 64, the weighted speech is given by

 (67)

where 0.68 is the pre-emphasis factor. Furthermore, for the open-loop pitch analysis, the computation is extended for a period of 8.75ms using the look-ahead from the future frame. This is done using the filter coefficients of the 4th subframe in the present frame. Note that this extended weighted signal is used only in the OL pitch analysis of the present frame.

If ACELP core is selected for WB, SWB or FB signals at bitrates higher than 13.2 kbps, its internal sampling rate is set to 16 kHz rather than 12.8 kHz. Nevertheless, the OL pitch analysis is done only at 12.8 kHz and the estimated OL pitch values are later resampled to 16 kHz. However, perceptually weighted input signal sampled at a16 kHz is still needed in the search of the adaptive codebook. The perceptual weighting filter at 16 kHz has the following form

 ()

where  and . Thus, for this case, the pre-emphasis is done as follows

 ()

The perceptual weighting at 25.6 kHz and 32 kHz is described later in this document.

#### 5.1.10.2 Correlation function computation

The correlation function for each of the three segments (or half-frames) is obtained using correlation values computed over a first pitch delay range from 10 to 115 (which has been decimated by 2) and over a second pitch delay range from 12 to 115 (which has been decimated by 2). Both of the two delay ranges are divided into four sections: [10, 16], [17, 31], [32, 61] and [62, 115] for the first delay range and [12, 21], [22, 40], [41, 77] and [77, 115] for the second delay range, so that the two sets of four sections overlap. The first sections in the two sets, [10, 16] and [12, 21] are, however, used only under special circumstances to avoid quality degradation for pitch lags below the lowest pitch quantization limit. Due to this special use of the first sections in the sets, omitting the pitch lags 10 and 11 in the second set of sections presents no quality issues. In addition, the second set omits pitch lags between 17 and 20, when the first sections are not used. The first section is mainly used in speech segments with stable, short pitch lags and the above limits have therefore a negligible effect on the overall pitch search and quantization performance.

The autocorrelation function is first computed on a decimated weighted signal for each pitch lag value in both sets by

 (70)

where the summation limit  depends on the delay section, i.e.:

 ()

This will ensure that, for a given delay value, at least one pitch cycle is included in the correlation computation. The autocorrelation window is aligned with the first sample of each of the two 10-ms segments of the current frame, where the autocorrelation can thus be calculated directly according to equation (70). To maximize the usage of the look-ahead segment, the autocorrelation window in the third segment is aligned with the last available sample. In this final segment the autocorrelation function of equation (70) is computed backwards, i.e., the values of are negative. Therefore, the computation as such is the same for all three segments, only the window alignment differs and the indexing of the signal is reversed in the last segment.

#### 5.1.10.3 Correlation reinforcement with past pitch values

The autocorrelation function is then weighted for both pitch delay ranges to emphasize the function for delays in the neighbourhood of pitch lag parameters determined in the previous frame (extrapolated pitch lags).

The weighting function is given by a triangular window of size 27 and it is centred on the extrapolated pitch lags. The weighting function is given by

 (72)

where is a scaling factor based on the voicing measure from the previous frame (the normalized pitch correlation) and the pitch stability in the previous frame. During voiced segments with smooth pitch evolution, the scaling factor is updated in each frame by adding a value of  and it is upper-limited to 0.7. is the average of the normalized correlation in the two half frames of the previous frame and is given in equation (). The scaling factor  is reset to zero (no weighting) if is less than 0.4 or if the pitch lag evolution in the previous frame is unstable or if the relative frame energy of the previous frame is more than a certain threshold. The pitch instability is determined by testing the pitch coherence between consecutive half-frames. The pitch values of two consecutive half-frames are considered coherent if the following condition is satisfied:



where  and  denote the maximum and minimum of the two pitch values, respectively. The pitch evolution in the current frame is considered stable if pitch coherence is satisfied for both, the first half-frame of the current frame and the second half-frame of the previous frame as well as the first half-frame and the second half-frame of the current frame.

The extrapolated pitch lag in the first half-frame,  , is computed as the pitch lag from the second half-frame of the previous frame plus a pitch evolution factor  , computed from the previous frame (described in subclause 5.1.10.7). The extrapolated pitch lag in the second half-frame, , is computed as the pitch lag from the second half-frame of the previous frame plus twice the pitch evolution factor. The extrapolated pitch lag in the look ahead, , is set equal to . That is

 ()

where  is the pitch lag in the second half-frame of the previous frame. The pitch evolution factor is obtained by averaging the pitch differences of consecutive half-frames that are determined as coherent (according to the coherence rule described above).

The autocorrelation function weighted around an extrapolated pitch lag  is given by

 ()

#### 5.1.10.4 Normalized correlation computation

After weighting the correlation function with the triangular window of equation (72) centred at the extrapolated pitch lag, the maxima of the weighted correlation function in each of the four sections (three sections, if the first section is not used) are determined. This is performed for both pitch delay ranges. Note that the first section is used only during high-pitched segments. For signals other than narrowband signals, this means that the open-loop pitch period of the second half-frame of the previous frame is lower than or equal to 34. For narrowband signals, the open-loop pitch period of the second half-frame of the previous frame needs to be lower than or equal to 24 and the scaling factor  has to be higher than or equal to 0.1. It is further noted that the scaling factor  is set to 0, if the previous frame were an unvoiced or a transition frame and the signal has a bandwidth higher than narrowband. In the following, the special case of three sections will not be explicitly dealt with if it arises directly from the text. The pitch delays that yield the maximum of the weighted correlation function will be denoted as , where *k* = 0,1,2,3 denotes each of the four sections. Then, the original (raw) correlation function at these pitch delays (pitch lags) is normalized as

 (75)

The same normalization is applied also to the weighted correlation function, , which yields . It is noted that  is aligned at the first sample of the corresponding half-frame for the two half-frames of the current frame and at the last sample of the look-ahead for the look-ahead segment, where the calculation is performed backwards in order to exploit the full look-ahead as well as possible.

At this point, four candidate pitch lags, , k = 0,1,2,3, have been determined for each of the three segments (two in the current frame and one in the look-ahead) in each of the two pitch delay ranges. In correspondence with these candidate pitch lags, normalized correlations (both weighted and raw) have been calculated. All remaining processing is performed using only these selected values, greatly reducing the overall complexity.

#### 5.1.10.5 Correlation reinforcement with pitch lag multiples

In order to avoid selecting pitch multiples within each pitch delay range, the weighted normalized correlation in a lower pitch delay section is further emphasized if one of its multiples is in the neighbourhood of the pitch lag in a higher section. That is,



where , is a voicing factor (normalized pitch correlation) from the previous frame, and  is the pitch value from the second half-frame of the previous frame. In addition, when the first section is searched and the pitch multiple of the shortest-section candidate lag is larger than 20 samples, the following reinforcement is performed:



Further, , is given by the voicing factor of the second half-frame in the previous frame if the normalized correlation in the second half-frame was stronger than in the first half-frame, or otherwise by the mean value of these two normalized correlations. In this way, if a pitch period multiple is found in a higher section, the maximum weighted correlation in the lower section is emphasized by a factor of 1.17. However, if the pitch lag in section 3 is a multiple of the pitch lag in section 2 and at the same time the pitch lag in section 2 is a multiple of the pitch lag in section 1, the maximum weighted correlation in section 1 is emphasized twice. This correlation reinforcement is, however, not applied at each section when the previous frame voicing factor, , is less than 0.6 and the pitch value is less than 0.4 times the previous pitch value (i.e., the pitch value does not appear to be a halved value of the previous frame pitch or larger). In this way, the emphasis of the correlation value is allowed only during clear voicing conditions or when the value can be considered to belong to the past pitch contour.

The correlation reinforcement with pitch lag multiples is independent in each of the two pitch delay ranges.

It can be seen that the "neighbourhood" is larger when the multiplication factor *k* is higher. This is to take into account an increasing uncertainty of the pitch period (the pitch length is estimated roughly with integer precision at a 6400 Hz sampling frequency). For the look-ahead part, the first line of the condition above relating to the highest pitch lags is modified as follows:



Note that first section is not considered in the correlation reinforcement procedure described here, i.e., the maximum normalized correlation in the first section is never emphasized.

#### 5.1.10.6 Initial pitch lag determination and reinforcement based on pitch coherence with other half-frames

An initial set of pitch lags is determined by searching for the maximum weighted normalized correlation in the four sections in each of the three segments or half-frames. This is done independently for both pitch delay ranges. The initial set of pitch lags is given by

 ()

where the superscript denotes the first, the second or the third (look‑ahead) half-frame.

To find the right pitch value, another level of weighting is performed on the weighted normalized correlation function, , in each section of each half-frame in each pitch delay range. This weighting is based on pitch coherence of the initial set of pitch lags,, with pitch lags, , *j* ≠ *i*, i.e., those from the other half-frames. The weighting is further reinforced with pitch lags selected from the complementary pitch delay range, denoted as , , *j* ≠ *i*. Further, the weighting favours section‑wise stability, where a stronger weighting is applied for coherent pitch values that are from the same section of the same set as the initial pitch lag, and a slightly weaker weighting is applied for coherent pitch values that are from a different section and/or a different pitch delay range than the initial pitch lag. That is, if the initial pitch lag in a half‑frameis coherent with a pitch lag of section *k* in half-frame , then the corresponding weighted normalized correlation of section in half-frameis further emphasized by weighting it by the value , if the initial pitch lag is also from section  in the same pitch delay range, or by , if the initial pitch lag is not from section *k* in the same pitch delay range. The variable is the absolute difference between the two analysed pitch lags and the two weighting factors are defined as



where is upper-bounded by 0.4, is upper-bounded by 0.25 and  is the raw normalized correlation (similar to the weighted normalized correlation, defined in equation ()). Finally,  is a noise correction factor added to the normalized correlation in order to compensate for its decrease in the presence of the background noise. It is defined as

 (77)

where  is the total background noise energy, calculated as described in subclause 5.1.11.1.

The procedure described in this subclause helps further to avoid selecting pitch multiples and insure pitch continuity in adjacent half-frames.

#### 5.1.10.7 Pitch lag determination and parameter update

Finally, the pitch lags in each half-frame, , and , are determined. They are selected by searching the maximum of the weighted normalized correlations, corresponding to each of the four sections across both pitch delay ranges. In case of VBR operation, the normalized correlations are searched in addition to the weighted normalized correlations for a secondary evaluation. When the normalized correlation of the candidate lag is very high (lower-bounded by 0.9) and it is considered a halved value (lower-bounded by a multiplication by 0.4 and upper-bounded by a multiplication by 0.6) of the corresponding candidate identified by searching the weighted normalized correlation, the secondary pitch lag candidate is selected instead of the firstly selected one.

In total, six or eight values are thus considered in each segment or half-frame depending on whether section 0 is searched. After determining the pitch lags, the parameters needed for the next frame pitch search are updated. The average normalized correlation is updated by:

 (78)

Finally, the pitch evolution factor  to be used in computing the extrapolated pitch lags in the next frame is updated. The pitch evolution factor is given by averaging the pitch differences of the consecutive half-frames that are determined as coherent. If  is the pitch lag in the second half of the previous frame, then pitch evolution is given by

 ()

Since the search is performed on the decimated weighted signal, the determined pitch lags, , and are multiplied by 2 to obtain the open-loop pitch lags for the three half-frames. That is

 (80)

In the following text, the following notation is used for the normalized correlations corresponding to the final pitch lags:

 (81)

#### 5.1.10.8 Correction of very short and stable open-loop pitch estimates

Usually, music harmonic signals or singing voice signals have short pitch lags and they are more stationary than normal speech signals. It is extremely important to have the correct and precise short pitch lags as incorrect pitch lags may have a serious impact upon the quality.

The very short pitch range is defined from  to  at the sampling frequency  kHz. As the pitch candidate is so short, pitch detection of using time domain only or frequency domain only solution may not be reliable. In order to reliably detect short pitch value, three conditions may need to be checked: (1) in frequency domain, the energy from 0 Hz to Hz must be relatively low enough; (2) in time domain, the maximum short pitch correlation in the pitch range from  to  must be relatively high enough compared to the maximum pitch correlation in the pitch range from  to ; (3) the absolute value of the maximum normalized short pitch correlation must be high enough. These three conditions are more important; other conditions may be added such as Voice Activity Detection and Voiced Classification.

Suppose notes the average normalized pitch correlation value of the four subframes in the current frame:

 ()

are the four normalized pitch correlations calculated for each subframe; for each subframe, the best pitch candidate is found in the pitch range from to . The smoothed pitch correlation from previous frame to current frame is

 ()

Before the real short pitch is decided, two pre-decision conditions are checked first : (a) check if the harmonic peak is sharp enough, which is indicated by the flag . It is used to decide if the initial open-loop pitch is correct or not; (b) check if the maximum energy in the frequency region [*0, FMIN*] is low enough, which is indicated by the flag .

(a) Determine base pitch frequency  according to the initial open-loop pitch 

 ()

Then, based on the amplitude spectrum of input signal in frequency domain, determine the decision parameters which are used to confirm whether the pitch related to the base pitch frequency is accurate. The decision parameters include energy spectrum difference, average energy spectrum and the ratio of energy spectrum difference and average energy spectrum.

Compute the energy spectrum difference and the average energy spectrum of the frequency bins around base pitch frequency 

 ()

 ()

Compute the weighted and smoothed energy spectrum difference and average energy spectrum

 ()

 ()

where  and  are weighted and smoothed energy spectrum difference and average energy spectrum of the frequency bins around the base pitch frequency.

Compute the ratio of energy spectrum difference and average energy spectrum

 ()

Based on the decision parameters calculated above, confirm whether the initial open-loop pitch is accurate.

The harmonic sharpness flag is determined as follows:

 ()

If the above conditions are not satisfied,  remains unchanged.

(b) Assume that the maximum energy in the frequency region  (Hz) is (dB) , the maximum energy in the frequency region  (Hz) is (dB), the relative energy ratio between and is given by

 ()

This energy ratio is weighted by multiplying an average normalized pitch correlation value ,

 ()

Before using the  parameter to detect the lack of low frequency energy, it is smoothed in order to reduce the uncertainty,

 ()

where the  is the low frequency smoothed energy ratio. If  then a lack of low frequency energy has been detected (otherwise not detected).  is determined by the following procedure,

 ()

If the above conditions are not satisfied,  remains unchanged.

An initial very short pitch candidate  is found by searching a maximum normalized pitch correlation from  to ,

 ()

If  notes the current short pitch correlation,

 ()

The smoothed short pitch correlation from previous frame to current frame is

 ()

By using all the available parameters, the final very short pitch lag is decided with the following procedure,

 ()

wherein  is a flag which forces the codec to select the time domain CELP coding algorithm for short pitch signal even if the frequency domain coding algorithm and AUDIO class is previously selected;  is a flag which forces the coder to select VOICED class for short pitch signal.

#### 5.1.10.9 Fractional open-loop pitch estimate for each subframe

The OL pitch is further refined by maximizing the normalized correlation function with a fractional resolution around the pitch lag values and (in the 12.8-kHz sampling domain). The fractional open-loop pitch lag is computed four times per frame, i.e., for each subframe of 64 samples. This is similar as the closed-loop pitch search, described in later in this specification. The maximum normalized correlation corresponding to the best fractional open-loop pitch lag is then used in the classification of VC frames (see subclause 5.1.13.2). The fractional open-loop pitch search is performed by first maximizing an autocorrelation function  of the perceptually weighted speech for integer lags in the interval [], where  for the search in the first and the second subframes, and for the third and fourth subframes. The autocorrelation function is similar to equation () except that perceptually weighted speech at 12.8 kHz sampling rate is used, i.e,

 ()

In the above equation, corresponds to the first sample in each subframe.

Let  be the integer lag maximizing . The fractional open-loop pitch search is then performed by interpolating the correlation function  and searching for its maximum in the interval . The interpolation is performed with a 1/4 sample resolution using an FIR filter – a Hamming windowed sinc function truncated at ±17. The filter has its cut-off frequency (–3 dB) at 5062 Hz and –6 dB at 5760 Hz in the 12.8 kHz domain. This means the interpolation filter exhibits a low-pass frequency response. Note that the negative fractions are not searched if  coincides with the lower end of the searched interval, i.e., if.

Once the best fractional pitch lag,, is found, the maximum normalized correlation is computed similarly to equation (), i.e.,

 ()

The same normalization is applied also to the weighted correlation function,, which yields .

At this point, four candidate pitch lags, , *k* = 0,1,2,3, have been determined for each of the three half-frames (two in the current frame and one in the look ahead) in each of the two pitch delay ranges. In correspondence with these candidate pitch lags, normalized correlations (both weighted and raw) have been calculated. All remaining processing is performed using only these selected values, greatly reducing the overall complexity.

Note that the last section (long pitch periods) in both pitch delay ranges is not searched for the look ahead part. Instead, the normalized correlation values and the corresponding pitch lags are obtained from the last section search of the second half-frame. The reason is that the summation limit in the last section is much larger than the available look ahead and also the computational complexity is reduced.

The fractional OL pitch estimation as described above is performed only for bitrates lower or equal to 24.4 kbps. For higher bitrates, the VC mode is not supported and consequently, there is no reason to estimate pitch with fractional resolution. Therefore, at higher bitrates,  where for the first and for the second subframe *i*=0 and for the third and the fourth subframe *i*=1.

### 5.1.11 Background noise energy estimation

The background noise energy is estimated (updated) in two stages. In the first stage, noise energy is updated only for critical bands where the current frame signal energy is less than the previously estimated background noise energy. This stage is called the downward noise energy update. In the second stage, noise energy is updated if the signal characteristics are statistically close to the model of background noise. Therefore, in the second stage, noise energy can be updated regardless of the current frame signal energy.

#### 5.1.11.1 First stage of noise energy update

The total noise energy per frame is computed as follows:

 (101)

where  is the estimated noise energy in the *i*th critical band of the previous frame.

The noise energy per critical band  is initialized to 0.0035 dB. The updated noise energy in the *i*th critical band, denoted , is computed as follows:

 (102)

where  corresponds to the energy per critical band calculated in the second spectral analysis in the previous frame, and  is the estimated noise energy per critical band also in the previous frame. Noise energy is then updated only in critical bands that have lower energy than the background noise energy. That is

 ()

The superscript [0] in the above equation is used to stress that it corresponds to the current frame.

Another feature used in noise estimation and SAD is an estimate of the frame to frame energy variation. The absolute energy difference between the current and the last frame is calculated,.

. ()

where the superscript [-1] has been used to denote the previous frame. The frame energy variation is then used to update the feature

 (105)

Other energy features that are updated before the SAD and the second stage of the noise estimation are first initialized during the very two frames after encoder initialization. The initialization is done as follows

 (106)

After the two frames of initialization the total frame energy is smoothed by means of LP filtering. That is:

 (107)

The features and are envelope tracking features of the frame energy and are used to create the long-term minimum energy and an estimate of the energy dynamics . That is

, (108)

To calculate the following processing is applied:

 (109)

where  is the number of frames since the last harmonic event from the previous frame. See clause 5.1.11.3.2 for details about its computation. The new value of is then used to update its long-term value through an AR process. That is

 (110)

where the parameter is set as follows

 ()

The energy dynamics feature  is just an LP-filtered version of the difference between and . That is

 (112)

#### 5.1.11.2 Second stage of noise energy update

In the second stage of the noise energy update, the critical bands not updated in the first stage are updated only if the current frame is inactive. However, the SAD decision obtained in clause 5.1.12, which is based on the SNR per critical band, is not used for determining whether the current frame is inactive and whether the noise energy is to be updated. Another decision is performed based on other parameters not directly dependent on the SNR per critical band. The basic parameters used for the noise update decision are:

– pitch stability

– signal non-stationarity

– normalized correlation (voicing)

– ratio between 2nd‑order and 16th‑order LP residual error energies

These parameters have generally low sensitivity to the noise level variations. Another set of parameters is calculated to cover harmonic (tonal) signals and, in particular, music. These parameters prevent the noise energy to be updated, when strong harmonicity or tonality is detected even when its energy is low. The parameters related to the detection of tonal signals are

– spectral diversity

– complementary non-stationarity

– HF energy

– tonal stability

The reason for not using the SAD decision for noise update is to make the noise estimation robust to rapidly changing noise levels. If the SAD decision was used for the noise update, a sudden increase in noise level would cause an increase of SNR even for inactive speech frames, preventing the noise estimator to update, which in turn would maintain the SNR high in the following frames. Consequently, the noise update would be blocked and some other logic would be needed to resume the noise adaptation.

##### 5.1.11.2.1 Basic parameters for noise energy update

The pitch stability counter is computed as

 ()

where *d*[0], *d*[1] and *d*[-1] are the OL pitch lags for the first half-frame, second half-frame and the second half-frame of the pervious frame. The pitch stability is true if the value of *pc* is less than 12. Further, for frames with low voicing, *pc* is directly set to 12 to indicate pitch instability. That is

if  then *pc* = 12, ()

where  are the normalized raw correlations as defined in clause 5.1.10.7 and *re* is a correction added to the normalized correlation in order to compensate for the decrease of normalized correlation in the presence of background noise, defined in clause 5.1.10.6. The voicing threshold *thCpc* = 0.52 for WB inputs, and *thCpc* = 0.65 for NB inputs.

Signal non-stationarity is analysed based on the product of ratios between the current frame energy per critical band and its long-term average per critical band. The average long-term energy per critical band is calculated as

, for *i* = *b*min to *b*max, (115)

where *b*min= 0 and *b*max= 19 in case of WB signals, and *b*min= 1 and *b*max= 16 in case of NB signals. The update factor  is a linear function of the relative frame energy, defined in clause 5.1.5.2 and it is given as follows

, constrained by  ()

where all negative values of  are replaced by 0. The frame non-stationarity is then given by the product of the ratios between the frame energy and its long-term average calculated in the previous frame. That is

 (117)

The voicing factor for noise update is given by

 ()

The ratio between the LP residual energy after 2nd‑order and 16th‑order analysis is given by

 ()

where *E*(2) and *E*(16) are the LP residual energies after 2nd‑order and 16th‑order analysis, and computed in the Levinson-Durbin recursion (see clause 5.1.9.4). This ratio reflects the fact that, to represent a signal spectral envelope, a higher order of LP is generally needed for speech signal than for noise. In other words, the ratio between *E*(2) and *E*(16) is expected to be lower for noise than for active speech.

##### 5.1.11.2.2 Spectral diversity

The basic parameters for noise estimation have their limitations for certain music signals, such as piano concerts or instrumental rock and pop. Spectral diversity gives information about significant spectral changes. The changes are tracked in the frequency domain in critical bands by comparing energies in the first spectral analysis of the current frame with the second spectral analysis two frames ago. The energy per critical band corresponding to the first spectral analysis of the current frame is denoted as  and is defined in clause 5.1.5.2. Let the energy per critical band corresponding to the second spectral analysis two frames ago be denoted as . For all bands higher than 9, the maximum and the minimum of the two energies is found as

, for *i* = 10,..,*b*max, ()

where *b*max= 19 in case of WB signals, and *b*max= 16 in case of NB signals. The energy ratio is the calculated as

, for *i* = 10,..,*b*max. ()

The spectral diversity is then calculated as the normalized weighted sum of the ratios in all critical bands with the weight itself being the maximum energy . That is

 ()

The spectral diversity is used as an auxiliary parameter for the complementary non‑stationarity described below.

##### 5.1.11.2.3 Complementary non-stationarity

The complementary non-stationarity is motivated by the fact that the non-stationarity described in clause 5.1.11.2.1 and calculated in equation (117) is low when a sharp energy attack in a harmonic signal is followed by a slow energy decay. In this case, the average long-term energy per critical band, , slowly increases after the attack whereas the current energy per critical band, , slowly decreases. At certain point (few frames after the attack frame) they are the same yielding only a small value of the *nonstat* parameter. This indicates to the noise estimation logic an absence of active signal which is wrong. It may lead to a false update of the background noise and consequently a collapse of the SAD algorithm.

To overcome this problem, there is an alternative calculation of the average long-term energy per critical band. It is calculated in the same way as in equation (115) but with a different factor. That is

, for *i* = *b*min to *b*max. ()

where  is initialized to 0.03. The update factor  and reset to 0 if *pdiv* > 5. The complementary non-stationarity parameter is then calculated in the same way as *nonstat* but using  instead of . That is:

 ()

The complementary non-stationarity must be used by the noise estimation logic only in certain signal passages. These are characterized by the parameter  which can be described as the average non-binary decision combined from non-stationarity and tonal stability. That is

if *nonstat* > *thstat* OR *ptonal* = 1 then  otherwise 

where  is in the range [0; 1] and *ptonal* is the tonal stability described in clause 5.1.11.2.5 and defined in equation (136).

##### 5.1.11.2.4 HF energy content

The HF energy content represents another parameter, which is used for the detection of certain noise‑like musical signals such as cymbals or low-frequency drums. This parameter is calculated as

, constrained by  ()

but only for frames that have at least a minimal HF energy, i.e. when both the numerator and the denominator of the above equation are higher than 100. If this is not fulfilled, . Finally, the long-term value if this parameter is calculated as

 ()

where  is initialized to zero.

##### 5.1.11.2.5 Tonal stability

The tonal stability exploits the harmonic spectral structure of certain musical signals. In the spectrum of such signals there are tones which are stable over several consecutive frames. To exploit this feature, it is necessary to track the positions and shapes of strong spectral peaks. The tonal stability is based on a correlation between the spectral peaks in the current frame and the past frame. The input to the algorithm is an average logarithmic energy spectrum, defined as

, , (127)

where  is defined in clause 5.1.5.2 and the superscripts [0] and [1] denote the first and the second spectral analysis, respectively. In the following text, the term "spectrum" will refer to the average logarithmic energy spectrum, as defined by the above equation.

The tonal stability is calculated in three stages. In the first stage, indices of local minima of the spectrum are searched in a loop and stored as *i*min. This is described by the following equation

,  , (128)

The index 0 is added to  if . Consequently, the index 127 is added to , if . Let us denote the total number of minima found as *N*min. The second stage consists of calculating a spectral floor and its subtraction from the spectrum. The spectral floor is a piece-wise linear function which runs through the detected local minima. Every piece between two consecutive minima  and  can be described by a linear function as

, , (129)

where *k* is the slope of the line and . The slope is calculated by

 (130)

Thus, the spectral floor is a logical connection of all pieces. The leading bins of the spectrum up to  and the terminating bins of the spectrum from  are set to the spectral values themselves, i.e.

 (131)

Finally, the spectral floor is subtracted from the spectrum by

,  (132)

and the result is the residual spectrum. The calculation of the spectral floor and its subtraction is illustrated in the following figure.



Figure 9 : Spectral floor in the tonal stability

The third stage of the tonal stability calculation is the calculation of the correlation map and the long-term correlation map. This is again a piece-wise operation. The correlation map is created on a peak-by-peak basis where each two consecutive minima delimit one peak. Let us denote the residual spectrum of the previous frame as . For every peak in the current residual spectrum, normalized correlation is calculated with the previous residual spectrum. The correlation operation takes into account all indices (bins) of that peak delimited by two consecutive minima, i.e.

,  (133)

where the leading bins up to  and the terminating bins from  are set to zero. The figure below shows a graphical representation of the correlation map.



Figure 10 : Correlation map in the tonal stability calculation

The correlation map of the current frame is used to update its long-term value, which can be expressed as

,  ()

where . If any value of  exceeds the threshold of 0.95, the flag *fstrong* is set to one, otherwise it is set to zero. The long-term correlation map is initialized to zero for all *k*. Finally, all bins of  are summed together by

 ()

In case of NB signals, the correlation map in higher bands is very low due to missing spectral content. To overcome this deficiency, *msum* is multiplied by 1.53.

The decision about tonal stability is taken by subjecting *msum* to an adaptive threshold *thtonal*. This threshold is initialized to 56 and it is updated in every frame by

if *msum* > 56 then *thtonal* = *thtonal* – 0.2 otherwise *thtonal* = *thtonal* + 0.2

and is upper limited by 60 and lower limited by 49. Thus, it decreases when the summed correlation map is relatively high, indicating a good tonal segment, and increases otherwise. When the threshold is lower, more frames will be classified as tonal, especially at the end of active music periods. Therefore, the adaptive threshold may be viewed as a hangover.

The *ptonal* parameter is set to one whenever *msum* is higher than *thtonal* or when the flag *fstrong* is set to one. That is:

if *msum* > *thtonal* OR *fstrong* = 1 then *ptonal* = 1 otherwise *ptonal* = 0 (136)

##### 5.1.11.2.6 High frequency dynamic range

From the residual spectrum as described in equation 116, another parameter is computed. This parameter is called the high frequency dynamic is derived from the high band spectral dynamic of the residual spectrum and is used to set the high frequency dynamic range flag which is used inside the GSC to decide about the number of subframe and the bit allocation. The high frequency dynamic is compute as the average of the last 40 bin from the residual spectrum:

 ()

And the high frequency dynamic range flag is set depending on the past values and the actual high frequency dynamic as :

 ()

Where represents the frame at time t and represents the average high frequency dynamic at when the last time the flag was set to 0.

##### 5.1.11.2.7 Combined decision for background noise energy update

The noise energy update decision is controlled through the logical combination of the parameters and flags described in the previous sections. The combined decision is a state variable denoted *pnup* which is initially set to 6, and which is decremented by 1 if an inactive frame is detected or incremented by 2 if an active frame is detected. Further, *pnup* is bounded by 0 and 6. The following diagram shows the conditions under which the state variable *pnup* is incremented by 2 in each frame.



Figure 11 : Incrementing the state variable for background noise energy update

where, for WB signals, *thsta* = 350000, *thCnorm* = 0.85 and *thresid* = 1.6, and for NB signals, *thsta* = 500000, *thCnorm* = 0.7 and *thresid* = 10.4. If *pnup* is not incremented in any of the conditions from the above diagram, it is automatically decremented by 1. Therefore, it takes at least 6 frames before *pnup* reaches 0 which signals the subsequent logic that background noise energy can be updated. The final decision about background noise energy update is described in the subsequent section.

#### 5.1.11.3 Energy-based parameters for noise energy update

The parameters in this section are used in addition to the  described in the previous section to control when it is possible and safe to allow the noise estimate sub-bands to be increased according to the pre calculated noise estimate calculated in equation (102).

##### 5.1.11.3.1 Closeness to current background estimate

Similar to  and the parameter  represents a spectral difference. The difference is that it is the closeness/variation compared to the current background noise estimate that is measured. The calculation of the feature also differs in calculation during initialization, that is , or during normal operation. During initialization the comparison is made using a constant,  which is the initialization value for the sub-band energies, as shown in

 ()

This is done to reduce the effect of decision errors in the background noise estimation during initialization. After the initialization period the calculation is made using the current background noise estimate of the respective sub-band, according to:

 ()

It is worth noting that the calculation of  is not dependent of the band width as it is made over the same sub-bands regardless of the input bandwidth.

##### 5.1.11.3.2 Features related to last correlation or harmonic event

Two related features are created which relate to the occurrence of frames where correlation or harmonic events are detected. The first is a counter, , that keeps track of how many frames that have passed since the last frame where correlation or harmonic event has occurred. That is if a correlation or harmonic event is detected the counter is reset otherwise it is incremented by one, according to:

 (141)

where  is the normalized correlation in the first or the second half-frame and  is the result of the tonal detection in clause 5.1.11.2.5. If the counter  is larger than 1 it is limited to 1 if or if  AND is 1. Depending on the estimated short term variance of the input frame energy the current value of the counter  can be reduced to one quarter of its value (or 1 if it was less than 4). The reduction is made for frames where  where  and the short therm variance estimate of the frame energy is larger than 8.0. The other feature is the long term measure of the relative occurrence of correlation or tonal frames. It is represented as a scalar value, , which is updated using a first order AR-process with different time constants depending on if the current frame is classified as a correlation/tonal frame or not according to:

 (142)

where the test, , represents a detection of a correlation/tonal event.

##### 5.1.11.3.3 Energy-based pause detection

To improve the tracking of the background noise the energy pause detector monitors the number of frames since the frame energy got close to the long-term minimum frame energy estimate. For inactive frames the counter is 0 or higher, , where positive integers represent the number of frames since the start of the current pause. When active content is detected the counter is set and kept at, . Initially =0 so the detector is in a inactive state and checks for an energy increase relative the long term minimum energy tracker  that could triggers a transition to and active state:

 ()

If the detector is in an active state the detector checks if the frame energy once again has come close to the long term minimum energy

 ()

The final step in the update of this parameter is to increment the counter if the detector is in an inactive state

 ()

##### 5.1.11.3.4 Long-term linear prediction efficiency

This section describes how the residual energies from the linear prediction analysis made in clause 5.1.9 can be used to create a long term feature that can be used to better determine when the input signal is active content or background noise based on the input signal alone.

The analysis provides several new features by analysing the linear prediction gain going from 0th-order to 2nd-order linear prediction and going from 2nd-order to 16th-order prediction. Starting with the 2nd order prediction residual energy that is compared to the 0th-order prediction residual energy, which is the energy of the input signal. For a more stable long term feature the gain is calculated and limited as

 ()

where is the energy of the input signal and  is the residual energy after the second-order linear prediction (see clause 5.1.9.4). The limited prediction gain is then filtered in two steps to create long term estimate of this gain. The first is made using

 ()

and typically this will become either 0 or 8 depending on the type of background noise in the input once there is a segment of background only input. A second feature is then created using the difference between the first long term feature and the frame by frame limited prediction gain according to:

. (148)

This will give an indication of the current frames prediction gain compared to the long term gain. This difference is used to create a second long term feature, this is done using a filter with different filter coefficient depending on if the long term difference is higher or lower than the currently estimated average difference according to

. (149)

This second long term feature is then combined with the frame difference to prevent the filtering from masking occasional high frame differences, the final parameter is the maximum of the frame and the long term version of the feature

. (150)

The feature created using the difference between 2nd order prediction and 16th order prediction is analysed slightly differently. The first step here is also to calculate prediction gain as

 (151)

where represents the residual energy after a 2nd order linear prediction and  is the residual energy after a 16th order linear prediction, see clause 5.1.9.4. This limited prediction gain is then used for two long term estimates of this gain, one where the filter coefficient differs if the long term estimate is to be increased or not as shown in

. (152)

The second long term estimate uses a constant filter coefficient, according to

. (153)

For most types of background signals both will be close to 0, but have different responses to content where the 16th order linear prediction is needed (typically for speech and other active content). The first  will usually be higher than the second. This difference between the long term features is measured according to

 (154)

which is used as an input to the filter which creates the third long term feature according to

. (155)

Also, this filter uses different filter coefficients depending on if the third long term signal is to be increased or not. Also here the long term signal is combined with the input signal to prevent the filtering from masking occasional high inputs for the current frame. The final parameter is then the maximum of the frame and the long term version of the feature

. (156)

Note that also some of the other calculated features in this sub section are used in the combination logic for the noise estimation, , , , , and.

##### 5.1.11.3.5 Additional long-term parameters used for noise estimation

Some additional parameters that processed to create long term estimates are three measures the relation of the current frames energy compared to the energy of the noise estimate. The first calculates the difference between the current frame energy and the level of the current noise estimate this is then filtered to build a long term estimate according to

 ()

Another feature estimates a long term estimate of how often the current frame energy is close to the level of the background estimate using:

 ()

The third estimate is a second order estimate for the number of frames that the current input has been close to the noise estimate. This is simply a counter is reset if the long term estimate  is higher than a threshold and incremented otherwise, as shown in

 ()

The last additional features calculates an long term estimate of the difference in the current frame energy to the long term minimum energy feature, this is done by low pass filtering the calculated energy difference according to

 ()

#### 5.1.11.4 Decision logic for noise energy update

Already in the first step of the noise estimation (see clause 5.1.11.1), the current noise estimate has been reduced in sub-bands where the background noise energy was higher than the sub-band energy for the current frame. The decision logic described in this subsection shows how it is decided when to update the background noise estimate and how large that update should be allowed to be by setting the step size, . The update is adapted based on the earlier described features or combinations thereof.

Every frame an attempt is made to adjust the background noise estimate upwards, where it is important not to do the update in active content. Several conditions are evaluated in order to decide if an update is possible and how large an allowed update should be. As it is always allowed to make downwards updates it is equally important that possible updates are not prevented for extended times as this will affect the efficiency of the SAD. The noise update uses a flag to keep track of the number of prevented noise updates, , the same flag is also used to indicate that no update has taken place. The counter  is initialized to the value 0 to indicate that no update has been done so far. When updates are successful it is set to 1 and for failed updates the counter is incremented by 1.

The major decision step in the noise update logic is whether an update is to be made or not and this is formed by evaluation of the following logical expression

 (161)

where  ensures that it is safe to do an update provided that any of the four pause detectors, , , , and  indicate that an update is allowed. Note that the last term in the condition  is not is not combined with  as it handles the noise estimation during initialization.

Starting with the mask which ensures that the normal updates only can occur when the current frame energy is close to the estimated long-term minimum energy, (see clause 5.1.11.1), is adjusted with a level dependent scaling of the estimated frame energy variations, , according to

 (162)

The first pause detector  is based on the metric control logic described in subclause 5.1.11.2.7, when is 0 updates are allowed, that is

 (163)

The second pause detector allows for updates for low energy frames if the estimated signal dynamics is high and a sufficient number of frames have passed since the last correlation event, that is

 ()

The third pause detector allows updates when there are consecutive frames that are similar in energy to the current low level frames in a row,

 ()

The last detector is itself a combination of a mask and two pause detectors and mainly uses the additional features described in subclause 5.1.11.3.4, the detector is evaluated using

 ()

where  is the mask for the detector and  and  are the additional detectors. For this detector the following seven flags are first evaluated. The first flag signals that the frame energy close to background noise energy where the threshold is adapted to the estimated frame to frame energy variations, as

 ()

The second flag signals a high linear prediction gain with 2nd order model for a stationary signal, and is defined as follows:

 (168)

The third flag signals that there is a low linear prediction gain for 16th order linear prediction

 ()

The fourth flag signals that the current frame has low spectral fluctuation

 ()

The fifth flag signals that the long term correlation is low

 ()

The sixth flag signals low long term correlation value including the current frame

 ()

The seventh and last flag signals a non-speech like input signal

 ()

Using the above flags it is possible to express the mask as

 ()

The two additional detectors  and , are also those built using sub detectors and additional conditions. Starting with  the sub detectors are:

 ()

where the combination metrics  and are combinations where the maximum of a number of metrics are used for the comparison

 ()

 ()

For the  sub detector

 ()

where the combination metric  is calculated as

 ()

The last term  in the  handles the special conditions of noise update during the initialization, which occurs during the 150 first frames after the codec start. Also the initialization flag is evaluated as a combination of two flags according to

 ()

where the first flag test for initialization period and a sufficient number of frames without correlation event, according to

 ()

The second flag evaluates a number of earlier calculated features against initialization specific thresholds according to

 ()

Every frame an attempt is made to adjust the background noise estimate upwards, as it is important not to do the update in active content several conditions are evaluated in order to decide if update is possible and how large an update that should be allowed. At the same time it is important that possible updates are not prevented for extended times. The noise update uses a flag to keep track of the number of prevented noise updates. The same flag is also used to indicate that no update has taken place. The flag  is initialized to the value 0 to indicate that no update has been done so far. When updates are successful it is set to 1 and for failed updates the counter is incremented by 1.

If the above condition is evaluated to 0, the noise estimation only checks if the current content might be music by evaluating the following condition

. (183)

If this is evaluated to 1 the sub-band noise level estimates are reduced. This is done to recover from noise updates made before or during music. The reduction is made per sub-band depending on if the current estimate is high enough, according to

. (184)

and is updated according to the definition in equation (198) before noise estimation is terminated for this frame.

The following steps are taken when  is evaluated to 1. First the step size, , is initially set to 0, before the process of determining if the noise update should be set to 1.0, 0.1, or 0.01. For the update  to be set to 1.0 the following condition

 ()

and any of the following conditions

 ()

 ()

 (188)

needs to be evaluated to 1. When this happens is also set to 1 before the noise estimation for the current frame is updated using the previously calculated new value, according to

, (189)

where is the pre-calculated new noise estimate from subclause 5.1.11.1. The noise estimation procedure is done for the current frame after the in equation (198) is updated.

If the above condition has failed then the  is set to 0.1 if any of the four following conditions are met

 ()

 ()

 (192)

. (193)

If the  has been set to 0.1 it will be reduced to 0.01 if

 (194)

and if the following condition is met

. (195)

If the  is set to 0.1 or 0.01, is set to 1 before the noise estimation for the current frame is made according to

, (196)

and the noise estimation procedure is done for the current frame after  is updated in equation (198).

If the conditions to set the  to 0.1 or 0.01 have failed, the step size is still 0 and noise update has potentially failed. After testing if the following condition is true

 (197)

the variable is incremented to keep track of potentially failed updates and the noise estimation is done after the following update of .

In all cases the noise estimation updates end with an update of  which is the long-term estimate of how frequent noise estimations could be possible according to

 (198)

and where  is calculated in clause 5.1.11.2.6.

### 5.1.12 Signal activity detection

In this module active signal is detected in each frame and the main flags for external use are the three flags,  and the combined . These flags are set to one for the active signal, which is any useful signal bearing some meaningful information. Otherwise, they are set to zero indicating an inactive signal, which has no meaningful information. The inactive signal is mostly a pause or background noise. The three flags represent different trade-offs between quality and efficiency, and are used respectively by various subsequent processing modules.

The entire signal activity detection (SAD) module described in this section consists of three sub-SAD modules. Two of the modules, namely the SAD1 and the SAD2, work on the spectral analysis of the 12.8kHz sampled signal, see subclause 5.1.12.1 and 5.1.12.2 respectively for detailed descriptions. The third module, namely the SAD3, operates on the CLDFB that runs on the input sampling frequency, see subclause 5.1.12.6. A preliminary activity decision, , is first obtained by combining two of the three sub-SAD modules, the SAD1 and SAD2, for input with bandwidth greater than NB, or directly from SAD1 for NB input. This preliminary decision is then further combined with the decision output  of the third sub-SAD module, the SAD3, depending upon the codec mode of operation and the input signal characteristic. The resulting decision is then feed to a DTX hangover module to produce the final output .

Internally the flag  is used to always produce a flag with DTX hangover whether DTX is on or off. When this no longer is needed and DTX is on  replaces the combined  to reduce the number of variables used externally.

#### 5.1.12.1 SAD1 module

The SAD1 module is a sub-band SNR based SAD with hangover that utilizes significance thresholds to reduce the amount off false detections for energy variations in the background noise. During SAD initialization period the following variables are set as follows

 (199)

The output of the SAD1 module is two binary flags (signal activity decisions)  and . The difference between them is due to the setting of parameters for the significance thresholds. The first binary decision  is used by the speech/music classification algorithm described in clause 5.1.13.5. The second binary decision  is developed further and leads to the final SAD1 decision, . Note that all decisions can be modified by the subsequent modules.

The spectral analysis described in clause 5.1.5 is performed twice per frame. Let and denote the energy per critical band for the first and second spectral analysis, respectively (as computed in clause 5.1.5.2). The average energy per critical band for the whole frame and part of the previous frame is computed as

 (200)

where denotes the energy per critical band from the second analysis of the previous frame,  and  hereafter denote respectively the minimum and the maximum critical band involved in the computation, where= 1, = 16 for NB input signals and = 0, = 19 for WB signals (see Table 2 in subclause 5.1.5.1). The signal-to-noise ratio (SNR) per critical band is then computed as

 (201)

where is estimated noise energy per critical band, as explained in clause 5.1.11.1. The average SNR per frame, in dB, is then computed using significance thresholds with two different settings

 (202)

where , , and  are control parameters that differ between codec modes and sampling rates.

Table 6: Control parameters for the significance thresholds for different bandwidths

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Bandwidth |  |  |  |  |
| NB | 2.65 | 0.05 | 1.75 | 0.25 |
| WB | 2.5 | 0.2 | 1.3 | 0.8 |
| SWB | 2.5 | 0.2 | 1.75 | 0.25 |

The signal activity is detected by comparing the two average SNR’s per frame to a certain threshold the first is then used without hangover and the second has a hangover period added to prevent frequent switching at the end of an active speech period. The threshold is a function of the long-term SNR and the estimated frame to frame energy variations, mainly the variation in noise but without the need to identify noise frames. The initial estimate of the long-term SNR is given by

 (203)

where is the long-term active signal energy, calculated in equation () and is the long-term noise energy, calculated in equation (). If this estimate is lower than the signal dynamics estimate  calculated in equation (). Then the estimate is adjusted according to

 ()

The energy variation is the  calculated in equation (105) in clause 5.1.11.1.

The threshold calculation is calculated in three steps, one initial value and two sequential modifications. The initial value is calculated as

 (205)

Where the function parameters,  are set according to the current input bandwidth summarized in the following table

Table 7: Functional parameters for the initial calculation for different bandwidths

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Bandwidth |  |  |  |  |
| NB | 0.1 | 16.0 | 4.00 | 1.15 |
| WB and SWB | 0.1 | 16.1 | 2.05 | 1.65 |

If the estimated SNR conditions are good, i.e. if , the threshold is updated and upper limited for certain low-level NB signals. That is

 (206)

##### 5.1.12.1.1 SNR outlier filtering

The average SNR per frame, , that is estimated as shown in equation (202) is updated such that any sudden instantaneous SNR variations in certain sub-bands do not cause spurious deviations in the average SNR from the long term behaviour. A set of bands and SNRs per band are determined and accumulated based on noise characteristics as shown in equations (209), (210). The critical band that contains the maximum average SNR is identified initially as the outlier band whose index is represented as, , and the outlier band SNR is given by,

 ()

 ()

The background noise energy is accumulated in bands  through  and in bands  through .

 ()

 ()

The average SNR, , is modified for WB and SWB signals through outlier filtering as follows,

(211)

The outlier filtering parameters used in updating the average SNR are listed in the table below.

Table 8: SNR outlier filtering parameters

|  |  |
| --- | --- |
| Parameter | value |
| MAX\_SNR\_OUTLIER\_1 | 10 |
| MAX\_SNR\_OUTLIER\_2 | 25 |
| MAX\_SNR\_OUTLIER\_3 | 50 |
| SNR\_OUTLIER\_WGHT\_1 | 1.0 |
| SNR\_OUTLIER\_WGHT\_2 | 1.01 |
| SNR\_OUTLIER\_WGHT\_3 | 1.02 |
| OUTLIER\_THR\_1 | 10 |
| OUTLIER\_THR\_2 | 6 |
| Maximum outlier band index  (MAX\_SNR\_OUTLIER\_IND) | 17 |
| TH\_CLEAN | 35 |

Based on the outlier band estimated in equation (207), a weighting is determined as per equation (211) and applied to SNRs per band (through outlier filtering by subtracting the SNR in the outlier band) or on the average SNR. The threshold, , is updated based on the outlier filtering and further statistics from background noise level variations, previous frame coder type, and the weighting of SNR per band. The threshold update is not performed when the long-term SNR, is below the clean speech threshold, TH\_CLEAN = 35dB.

 (212)

where the smoothed average SNR, , is calculated after the SNR outlier filtering is performed in equation ().

 ()

The updated threshold, , as shown in equation () and the updated average SNR, , as shown in equation () are used in signal activity detection logic as described in Clause 5.1.12.3.

#### 5.1.12.2 SAD2 module

The SAD2 module is also a sub-band SNR based SAD and makes an activity decision for each frame by measuring the frame’s modified segmental SNR. The output of SAD2 module is a binary flag  which is set to 1 for active frame and set to 0 for inactive frame. For each frame, the SNR per critical band is first computed. The average energy per critical band for the whole frame and part of the previous frame is computed as

 ()

where denotes the energy per critical band from the second spectral analysis of the previous frame, and denote respectively the energy per critical band for the first and second spectral analysis of the current frame, = 0, = 19. More weighting is given to the energy of the second spectral analysis for the current frame if the energy of the second spectral analysis is higher than the first spectral analysis. This is designed to improve the detection of signal onsets. The SNR per critical band is then computed as

 ()

where  is the estimated noise energy per critical band, as described in clause 5.1.11. The SNR per critical band is then converted to a logarithmic domain as

 ()

The log SNR per critical band is then modified by

 ()

where is the modified SNR per critical band,  is an offset value which is a function of the critical band and the long-term SNR of the input signal as calculated in equation (203), summation of  is constrained to be not greater than 2, and  is an exponential factor used to re-shape the mapping function between  and , is also a function of the long-term SNR of the input signal. The offset value  is determined as shown in the following table

Table 9: Determination of 

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | <2 | 2<7 | 7<18 | 18 |
| >24 | 0 | 0 | 0 | 0 |
| 18<24 | 0.1 | 0.2 | 0.2 | 0.2 |
| 18 | 0.2 | 0.4 | 0.3 | 0.4 |

and is determined as

 ()

The modified segmental SNR is then computed as

 ()

and a relaxed modified segmental SNR is also computed. The procedure of calculating the relaxed modified segmental SNR is similar to the calculation of the modified segmental SNR with the only difference being that, besides , another offset value  is also added to the log SNR per critical band  when calculating the relaxed modified SNR per critical band. The relaxed modified SNR per critical band is therefore computed as

 ()

where  is a function of the critical band and is determined as

 ()

The relaxed modified segmental SNR is used in the hangover process at a later stage of the algorithm.

A further enhancement (increase in value) is made to the modified segmental SNR if an unvoiced signal is detected. Unvoiced signal is detected if both of the two critical bands covering the highest frequency range have SNRs greater than a threshold of 5, i.e. if  and . In this case, the contributions of the overall modified segmental SNR from the two critical bands is boosted. The boost is performed over the two critical bands where the number of critical bands is extended from two to eight and the corresponding modified segmental SNR is re-computed over the extended bands as

 ()

where multiplication by 20/26 effectively performs the mapping of the modified segmental SNR calculated on the extended scale back onto the same scale as if it were computed over the original 20 critical bands. The re-computation of modified segmental SNR is only conducted if the computed value is greater than before. If no unvoiced signal of above type is detected, , which is the number of critical bands whose SNR is greater than a threshold of 2 is determined. If >13, a second type unvoiced signal is detected, and if the long-term SNR of the input signal is further below a threshold of 24, the modified segmental SNR in this case is re-computed as

 ()

where  and is limited to be a positive value.

The primary signal activity decision is made in SAD2 by comparing the modified segmental SNR to a decision threshold . The decision threshold is a piece wise linear function of the long-term SNR of the input signal and is determined as

 ()

If the modified segmental SNR is greater than the decision threshold, the activity flag  is set to 1, and a counter of consecutive active frames, , used by SAD2 is incremented by 1, and if the current frame is ineither a soft or a hard hangover period as described later in this subclause, the corresponding hangover period elapses by 1. Otherwise, the consecutive active frames counter  is set to 0 and the setting of is further evaluated by a hangover process.

The hangover scheme used by SAD2 consists of a soft hangover process followed by a hard hangover process. The soft hangover is designed to prevent low level voiced signals during a speech offset from being cut. When within the soft hangover period, the SAD2 is operating in an offset working state where the relaxed modified segmental SNR calculated earlier is used to compare to the decision threshold(compared to the normal working state where the modified segmental SNR is used). If the relaxed modified segmental SNR is greater than the decision threshold, the activity flag  is set to 1 and the soft hangover period elapses by 1. Otherwise, if the relaxed modified segmental SNR is not greater than the decision threshold, the soft hangover period is quit and the setting of  is finally evaluated by a hard hangover process. When within the hard hangover period, the activity flag  is forced to 1 and the hard hangover period elapses by 1. The soft hangover period is initialized if the number of consecutive voiced frames exceeds 3. The frame is considered a voiced frame if the pitch correlation is not low and the pitch stays relatively stable, that is, if  and  where ,, are respectively the normalized pitch correlation for the second half of the previous frame, the first half of the current frame and the second half of the current frame as calculated,  is the noise correction factor, ,,, are respectively the OL pitch lag for the second half of the previous frame, the first half of the current frame, the second half of the current frame and the look-ahead as described in subclause 5.1.10. The value to which the soft hangover period is initialized is a function of the long-term SNR of the input signal and the noise fluctuation  computed in equation (), and is determined as

Table : Determination of the soft hangover period initialization length

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
|  | 1 | 1 | 2 |
|  | 1 | 3 | 4 |

The hard hangover period is initialized if the consecutive active frames counter  reaches a threshold of 3. The value to which the hard hangover period is initialized is also a function of the long-term SNR of the input signal and the noise fluctuation, and is determined as

Table : Determination of the hard hangover period initialization length

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
|  | 1 | 1 | 2 |
|  | 1 | 1 | 3 |

The noise fluctuation  is estimated over background frames declared as inactive by the final SAD flag of SAD2, , by measuring the moving average of the segmental SNR in the logarithm domain. The noise fluctuation is computed as

 (225)

where  denotes the noise fluctuation of the previous frame, is the forgetting factor controlling the update rate of the moving average filter and is set to 0.99 for an increasing update ( when ) and 0.9992 for a decreasing update ( when ).  is constrained by  for decreasing updates and for increasing updates. To speed up the initialization of noise fluctuation, for the first 50 background frames, is set to 0.9 for increasing updates and 0.95 for decreasing updates,  is constrained by  for decreasing updates and for increasing updates.

#### 5.1.12.3 Combined decision of SAD1 and SAD2 modules for WB and SWB signals

The decision of the SAD1 module is modified by the decision of the SAD2 module for WB and SWB signals.

For  the decision logic is direct if the average SNR per frame is larger than the SAD decision threshold and if the final SAD2 flag is set to 1. That is,

 ()

Likewise, for  the decision logic is direct if the average SNR per frame is larger than the SAD decision threshold and if the final SAD2 flag is set to 1.That is,

 ()

otherwise,  is set to 0 and the hangover logic decides if  should be set to active or not.

The hangover logic works as a state machine that keeps track of the number of frames since the last active primary decision and if a sufficient number of consecutive active frames have occurred in a row to allow the final decision to remain active even if the primary decision has already gone inactive. Thus, if there has not been a sufficient number of primary decisions in a row there is no hangover addition and the final decision is set to inactive, that is  is set to 0 if  is 0.

The hangover length depends on , initially set to 0 frames and if  it is set to 4 frames and if it is set to 3 frames. The counting of hangover frames is reset only if at least 3 consecutive active speech frames () were present, meaning that no hangover is used ifin only one or two adjacent frames. This is to avoid adding the hangover after short energy bursts in the acoustic signal, increasing the average data rate in the DTX operation.

#### 5.1.12.4 Final decision of the SAD1 module for NB signals

Similarly to the WB case two primary decisions are generated but in this case there is no dependency on the SAD2 module. If  the frame is declared as active and the primary SAD flag,  is set to1. Otherwise, is set to 0.

To get the final SAD decision  there is a difference in how the primary decision is made and how the hangover is handled compared to WB. For NB signals the hangover has a window of 8 frames after the last run of three consecutive active primary decision, that is . During this hangover period the SAD decision is not automatically set to active; instead, the threshold is decreased by 5.2 if, and by 2 if . The SAD decision is then made by comparing the average SNR to the corrected threshold following condition (206). Again, counting of hangover frames is reset only if at least 3 consecutive active frames were present.

#### 5.1.12.5 Post-decision parameter update

After the final decision is formed, some SAD1-related long term-parameters are updated according to the primary and final decisions.

 ()

These are then used to form a measure of the long term primary activity of 

 ()

Similar metrics are generated for the 

 ()

These are then used to form an measure of the long term primary activity of 

 ()

To keep track of the history  decisions the registers  are updated so that  keeps track of the latest 50 frames with regard to the  decisions by removing the oldest decision and adding the latest and updating  so that it reflects the current number of active frames in the registers.

Similarly to keep track of the history for  decisions the registers  are updated so that  keeps track of the latest 16 frames with regard to the  decisions by removing the oldest decision and adding the latest and updating  so that it reflects the current number of active frames in the registers.

When the SAD decisions have been made, the speech music classifier decision has been made and the noise estimation for the current frame has been completed the long term estimates of active speech level, , and long term noise level estimate can be updated. During the four first frames the initialization is made for both variables using mainly the current input as follows

 ()

Where  is the total sub band noise level after update for the current frame. For the long term noise level estimate the initialization uses different filter coefficient during the remainder of the 150 frames initialization as follows

 (233)

For the active speech level the update after the initial four frames only occurs if  is 1 AND the  from the speech music classifier is 0. Where  is generated as  based on the features  calculated based on equation (329) in subclause 5.1.13.6.3. If those conditions are met then the speech level estimate is updated according to

 (234)

#### 5.1.12.6 SAD3 module

The SAD3 module is shown in Figure 12. The processing steps are described as follows:

1. Extract features of the signal according to the sub-band signals from CLDFB.
2. Calculate some SNR parameters according to the extracted features of the signal and make a decision of background music.
3. Make a pre-decision of SAD3 according to the features of the signal, the SNR parameters, and the output flag of the decision of background music and then output a pre-decision flag.
4. The output of SAD3 is generated through the addition of SAD3 hangover.



Figure 12: Block diagram of SAD3

##### 5.1.12.6.1 Sub-band FFT

Sub-band FFT is used to obtain spectrum amplitude of signal. Let *X[k, l]* denote the output of CLDFB applied to the *lth* sample in the *k th* sub-band. *X[k,l]* is converted into a frequency domain representation from the time domain by FFT as follows:

 (235)

The spectrum amplitude of each sample is computed in the following steps:

Step 1: Compute the energy of  as follows:

 ()

where ,  are the real part and the imaginary part of , respectively.

Step 2: If *k* is even, the spectrum amplitude, denoted by , is computed by

  ()

If *k* is odd, the spectrum amplitude is computed by

  ()

##### 5.1.12.6.2 Computation of signal features

In Pre-decision Energy Features (EF), Spectral Centroid Features (SCF), and Time-domain Stability Features (TSF) of the current frame are computed by using the sub-band signal; Spectral Flatness Features (SFF) and Tonality Features (TF) are computed by using the spectrum amplitude.

5.1.12.6.2.1 Computation of EF

The energy features of the current frame are computed by using the sub-band signal. The energy of background noise of the current frame, including both the energy of background noise over individual sub-band and the energy of background noise over all sub-bands, is estimated with the updated flag of background noise, the energy features of the current frame, and the energy of background noise over all sub-bands of the previous frame. The energy of background noise of the current frame will be used to compute the SNR parameters of the next frame (see subclause 5.1.12.6.3). The energy features include the energy parameters of the current frame and the energy of background noise. The energy parameters of the frame are the weighted or non-weighted sum of energies of all sub-bands.

The frame energy is computed by:

 ()

The energy of sub-band divided non-uniformly is computed by:

 ()

Where  is the sub-band division indices of . The sub-bands based on this kind of division are also called SNR sub-bands and are used to compute the SNR of sub-band.  is the number of SNR sub-bands.

The energy of sub-band background noise of the current frame is computed by:

 (241)

Where  is the energy of sub-band background noise of the previous frame.

The energy of background noise over all sub-bands is computed according to the background update flag, the energy features of the current frame and the tonality signal flag, and it is defined as follows:

 ()

If certain conditions that include at least that the background update flag is 1 and the tonality signal flag  is 0 are met,  and  are computed by:

 ()

 ()

Otherwise,  and  are computed by:

 ()

 ()

Where  and  are the sum of  and the counter of , respectively. The superscript [-1] denotes the previous frame and [0] denotes the current frame.

5.1.12.6.2.2 Computation of SCF

The spectral centroid features are the ratio of the weighted sum to the non-weighted sum of energies of all sub-bands or partial sub-bands, or the value is obtained by applying a smooth filter to this ratio. The spectral centroid features can be obtained in the following steps:

a) Divide the sub-bands for computing the spectral centroids as shown in Table 12.

Table 12: Sub-band division for computing spectral centroids

|  |  |  |
| --- | --- | --- |
| Spectral centroid feature number (i) |  |  |
| 2 | 0 | 9 |
| 3 | 1 | 23 |

b) Compute two spectral centroid features, i.e.: the spectral centroid in the first interval and the spectral centroid in the second interval, by using the sub-band division for computing spectral centroids in Step a) and the following equation:

 (247)

c) Smooth the spectral centroid in the second interval,  , to obtain the smoothed spectral centroid in the second interval by

 (248)

5.1.12.6.2.3 Computation of SFF

Spectral Flatness Features are the ratio of the geometric mean to the arithmetic mean of certain spectrum amplitude, or this ratio multiplied by a factor. The spectrum amplitude, is smoothed as follows:

 ()

where  and  are the smoothed spectrum amplitude of the current frame and the previous frame, respectively.  is the number of spectrum amplitude.

Then the smoothed spectrum amplitude is divided into three frequency regions as shown in Table 13 and the spectral flatness features are computed for these frequency regions.

Table 13: Sub-band division for computing spectral flatness

|  |  |  |
| --- | --- | --- |
| Spectral flatness number  (k) |  |  |
| 0 | 5 | 19 |
| 1 | 20 | 39 |
| 2 | 40 | 64 |

The spectral flatness features are the ratio of the geometric mean to the arithmetic mean of the spectrum amplitude or the smoothed spectrum amplitude.

Let +1 be the number of the spectrum amplitudes used to compute the spectral flatness feature. We have

 (250)

The spectral flatness features of the current frame are further smoothed as follows:

 (251)

Where and  are the smoothed spectral flatness features of the current frame and the previous frame respectively.

5.1.12.6.2.4 Computation of TSF

The time-domain stability features are the ratio of the variance of the sum of energy amplitudes to the expectation of the squared sum of energy amplitudes, or this ratio multiplied by a factor. The time-domain stability features are computed with the energy features of the most recent N frame. Let the energy of the nth frame be . The energy amplitude of  is computed by

 (252)

By adding together the energy amplitudes of two adjacent frames from the current frame to the Nth previous frame, N/2 sums of energy amplitudes are obtained as

 (253)

Where is the energy amplitude of the current frame for *k* = 0 and the energy amplitude of the previous frames for *k* < 0.

Then the ratio of the variance to the average energy of the N/2 recent sums is computed and the time-domain stability  is obtained as follows:

 (254)

Note that the value of N is different when computing different time-domain stabilities.

5.1.12.6.2.5 Computation of TF

The tonality features are computed with the spectrum amplitudes. More specifically, they are obtained by computing the correlation coefficient of the amplitude difference of two adjacent frames, or with a further smoothing of the correlation coefficient, in the following steps:

a) Compute the spectrum-amplitude difference of two adjacent spectrum amplitudes in the current frame. If the difference is smaller than 0, set it to 0.

 (255)

b) Compute the correlation coefficient between the non-negative amplitude difference of the current frame obtained in Step a) and the non-negative amplitude difference of the previous frame to obtain the first tonality features as follows:

 (256)

where  is the amplitude difference of the previous frame.

Various tonality features can be computed as follows:

 (257)

where  are tonality features of the previous frame.

##### 5.1.12.6.3 Computation of SNR parameters

The SNR parameters of the current frame are computed with the background energy estimated from the previous frame, the energy parameters and the energy of the SNR sub-bands of the current frame.

The SNR of all sub-bands is computed by:

 (258)

The average total SNR of all sub-bands is computed by:

 (259)

where N is number of the most recent frames and  is of the ith frame.

The frequency-domain SNR is computed by:

 (260)

where  is the number of SNR sub-band and is the SNR of the ith sub-band by:

 (261)

The first long-time SNR is computed by:

 (262)

The computation method of and  can be found in subclause 5.1.12.6.6.

The second long-time SNR is obtained by accordingly adjusting a parameter  associated with  as follows:

 (263)

where:

 (264)

where  is the long-time background spectral centroid. If the current frame is active frame and the background-update flag is 1, the long-time background spectral centroid of the current frame is updated as follows:

 (265)

where  is the long-time background spectral centroid of the previous frame.

The initial long-time frequency-domain SNR of the current frame  is computed by:

 (266)

where  and  are respectively the frequency-domain SNR accumulator and frequency-domain SNR  counter when the current frame is pre-decided as active sound, and  and  are respectively  accumulator and  counter when the current frame is pre-decided as inactive sound. The superscript [-1] denotes the previous frame. The details of computation can be found in Steps e) and i) of subclause 5.1.12.6.6.

The smoothed average long-time frequency-domain SNR is computed by:

 (267)

The long-time frequency-domain SNR is computed by:

 (268)

where MAX\_LF\_SNR is the maximum of  .

##### 5.1.12.6.4 Decision of background music

With the energy features, , , , and  of the current frame, the tonality signal flag of the current frame is computed and used to determine whether the current frame is tonal signal. If it is a tonal signal, the current frame is music and the following procedure is carried out:

a) Suppose the current frame is a non-tonal signal, and a flag  is used to indicate whether the current frame is a tonal frame. If  = 1, the current frame is a tonal frame. If  = 0, the current frame is a non-tonal frame.

b) If >0.6 or its smoothed value  is greater than 0.86., go to Step c). Otherwise, go to Step d).

c) Verify the following three conditions:

(1) The time-domain stability feature  is smaller than 0.072;

(2) The spectral centroid feature  is greater than 1.2;

(3) One of three spectral flatness features is smaller than its threshold, .

If all the above conditions are met, the current frame is considered as a tonal frame and the flag  is set to 1. Then go to Step d).

d) Update the tonal level feature  according to the flag . The initial value of  is set in the region [0, 1] when the active-sound detector begins to work.

 (269)

Where  and  are respectively the tonal level of the current frame and the previous frame.

e) Determine whether the current frame is a tonal signal according to the updated  and set the tonality signal flag .

If  is greater than 0.5, the current frame is determined as a tonal signal. Otherwise, the current frame is determined as a non-tonal signal.

 (270)

##### 5.1.12.6.5 Decision of background update flag

The background update flag is used to indicate whether the energy of background noise is updated and its value is 1 or 0. When this flag is 1, the energy of background noise is updated. Otherwise, it is not updated.

The initial background update flag of the current frame is computed by using the energy features, the spectral centroid features, the time-domain stability features, the spectral flatness features, and the tonality features of the current frame. The initial background update flag is updated with the VAD decision, the tonality features, the SNR parameters, the tonality signal flag, and the time-domain stability features of the current frame to obtain the final background update flag. With the obtained background update flag, background noise is detected.

First, suppose the current frame is background noise. If any one of the following conditions is met, the current frame is not noise signal.

a) The time-domain stability  > 0.12;

b) The spectral centroid  > 4.0 and the time-domain stability  > 0.04;

c) The tonality feature  > 0.5 and the time-domain stability  > 0.1;

d) The spectral flatness of each sub-band or the average obtained by smoothing the spectral flatness is smaller than its specified threshold, or one of three spectral flatness features is smaller than its threshold: ;

e) The energy of the current frame  is greater than a specified threshold: , where  is the long time smoothed energy of the previous frame and  of kth frame is computed : ;

f) The tonality features  are greater than their corresponding thresholds: >0.60 **OR** >0.86;

g) The initial background update flag can be obtained in Steps a) - f). The initial background update flag is then updated. When the SNR parameters, the tonality features, and the time-domain stability features are smaller than their corresponding thresholds :  <0.3 **AND** <1.2 **AND**<0.5 **AND** <0.1 and both the combined  and  are set to 0, the background update flag is updated to 1.

##### 5.1.12.6.6 SAD3 Pre-decision

The SAD3 decision is computed with the tonality signal flag, the SNR parameters, the spectral centroid features, and the energy features. The SAD3 decision is made in the following steps:

a) Obtain the second long-time SNR  by computing and adjusting the ratio of the average energy of long-time active frames to the average energy of long-time background noise for the previous frame;

b) Compute the average of for a number of recent frames to obtain ;

c) Compute the SNR threshold for making SAD3 decision, denoted by, with the spectral centroid features  , the second long-time SNR , the long-time frequency-domain SNR , the number of previous continuous active frames  , and the number of previous continuous noise frames . Set the initial value of to. First, adjust  with the spectral centroid features, if the spectral centroids are located in the different regions, an appropriate offset may be added to. Then,  is further adjusted according to  , , , and . When  is greater than its threshold, the SNR threshold is appropriately decreased. When  is greater than its threshold, the SNR threshold is appropriately increased. If  is greater than a specified threshold, the SNR threshold may be accordingly adjusted.

d) Make an initial VAD decision with the SAD3 decision threshold  and the SNR parameters such as  and  of the current frame. First  is set to 0. If  >, or  ,  is set to 1. The initial VAD decision can be used to compute the average energy of long-time active frames  . The value of  is used to make SAD3 decision for the next frame.

 (271)

where  and  is computed by:

 (272)

 (273)

e) Update the initial SAD3 decision according to the tonality signal flag, the average total SNR of all sub-bands, the spectral centroids, and the second long-time SNR. If the tonality signal flag  is 1,  is set to 1. The parameters  and  are updated by:

 (274)

 (275)

If  >(B+  \*A), where A and B are two constants,  is set to 1. If any one of the following conditions is met:

condition 1: 

condition 2: 

 is to 1. Where ,  and  are the thresholds.

f) Update the number of hangover frames for active sound according to the decision result, the long-time SNR, and the average total SNR of all sub-bands for several previous frames, and the SNR parameters and the SAD3 decision for the current frame; See subclause 5.1.12.6.7 for details;

g) Add the active-sound hangover according to the decision result and the number of hangover frames for active sound of the current frame to make the SAD3 decision;

h) Make a combined decision with  and . The output flag of the combined decision is namely combined. See subclause 5.1.12.7;

i) After Steps g) and h), the average energy of long-time background noise, denoted by, can be computed with the SAD decisions combined  and . is used to make the SAD decision for the next frame. If both combined  and  are 0, ,  are updated and is computed as follows:

 (276)

 (277)

 (278)

where  and  is computed by:

 (279)

 (280)

The functions of the Pre-decision module are described in Steps a) - e) in this subclause.

##### 5.1.12.6.7 SAD3 Hangover

The long-time SNR and the average total SNR of all sub-bands are computed with the sub-band signal (See subclause 5.1.12.6.2.1 and 5.1.12.6.3). The current number of hangover frames for active sound is updated according to the SAD3 decision of several previous frames, ,  , other SNR parameters, and the SAD3 decision of the current frame. The precondition for updating the current number of hangover frames for active sound is that the flag of active sound indicates that the current frame is active sound. If both the number of previous continuous active frames <8 and <4.0, the curent number of hangover frames for active sound is updated by subtracting  from the minimum number of continuous active frames. Suppose the minimum number of continuous active frames is 8. The updated number of hangover frames for active sound, denoted by , is computed as follows:

 (281)

Otherwise, if both  > 0.9 and  > 50, the number of hangover frames for active sound is set according to the value of . Otherwise, this number of hangover frames is not updated.

 is set to 0 for the first frame. When the current frame is the second frame and the subsequent frames,  is updated according to the previous combined  as follows:

If the previous combined  is 1,  is increased by 1;

If the previous combined  is 0,  is set to 0.

#### 5.1.12.7 Final SAD decision

The feature parameters mentioned above are divided into two categories. The first feature category includes the number of continuous active frames, the average total SNR of all sub-bands, and the tonality signal flag .  is the average of SNR over all sub-bands for a predetermined number of frames. The second feature category includes the flag of noise type, the smoothed average long-time frequency-domain SNR  in a predetermined period of time, the number of continuous noise frames, frequency-domain SNR.

First, the parameters in the first and second feature categories and  and  are obtained. The first and second feature categories are used for the SAD detection.

The combined decision is made in the following steps:

1. Compute the energy of background noise over all sub-bands for the previous frame with the background update flag, the energy parameters, and the tonality signal flag of the previous frame and the energy of background noise over all sub-bands of the previous 2 frames. Computing the background update flag is described in subclause 5.1.12.6.5.
2. Compute the above-mentioned  with the energy of background noise over all sub-bands of the previous frame and the energy parameters of the current frame.
3. Determine the flag of noise type according to the above-mentioned parameters  and . First, the noise type is set to non-silence. Then, when  is greater than the first preset threshold and  is greater than the second preset threshold, the flag of noise type is set to silence.

Then, the features in the first and second feature categories,  and  are used for active-sound detection in order to make the combined decision of SAD.

When the input sampling frequency is 16 kHz and 32 kHz, the decision procedure is carried out as follows:

a) Select  as the initial value of the combined;

b) If the noise type is silence, and the frequency-domain SNR  is greater than 0.2 and the combined  set 0,  is selected as the output of the SAD, combined  . Otherwise, go to Step c).

c) If the smoothed average long-time frequency-domain SNR is smaller than 10.5 or the noise type is not silence, go to Step d). Otherwise, the initial value of the combined  in Step a) is still selected as the decision result of the SAD;

d) If any one of the following conditions is met, the result of a logical operation **OR** of  and  is used as the output of the SAD. Otherwise, go to Step e):

Condition 1: The average total SNR of all sub-bands is greater than the first threshold, e.g. 2.2;

Condition 2: The average total SNR of all sub-bands is greater than the second threshold, e.g. 1.5 and the number of continuous active frames is greater than 40;

Condition 3: The tonality signal flag is set to 1.

e) When the input sampling frequency is 32 kHz: If the noise type is silence,  is selected as the output of the SAD and the decision procedure is completed. Otherwise, the initial value of the combined  in Step a) is still selected as the decision result of the SAD. When the input sampling frequency is 16 kHz:  is selected as the output of the SAD and the decision procedure is completed.

When the input sampling frequency is neither 16 kHz nor 32 kHz, the procedure of the combined decision is performed as follows:

a) Select  as the initial value of the combined;

b) If the noise type is silence, go to Step c). Otherwise, go to Step d);

c) If the smoothed average long-time frequency-domain SNR is greater than 12.5 and =0, the combined  is set to. Otherwise, the initial value of combined  in Step a) is selected as the decision result of the SAD;

d) If any one of the following conditions is met, the result of a logical operation **OR** of  and  is used as the output of the final SAD, combined . Otherwise, the initial value of combined  in Step a) is selected as the decision result of the SAD;

Condition 1: The average total SNR of all sub-bands is greater than 2.0;

Condition 2: The average total SNR of all sub-bands is greater than 1.5 and the number of continuous active frames is greater than 30;

Condition 3: The tonality signal flag is set to 1.

After the combined  is obtained by using the above-mentioned method, it needs to be modified as follows:

a) Compute the number of background-noise updates,  according to the background update flag, specifically:

When the current frame is indicated as background noise by the background update flag and  is smaller than 1000,  increases by 1. Note that  is set to zero at the initialization of the codec.

b) Compute number of modified frames for active sound,  according to the SAD3 decision , the number of background-noise updates , and the number of hangover frames for active sound , specifically:

When the current frame is indicated as active sound by  and  is smaller than 12,  is selected as *max*(20, ).

c) Compute the final decision of SAD for the current frame according to the number of modified frames for active sound  and the combined , specifically:

When the current frame is indicated as inactive sound by the combined  and  is greater than 0, the final decision of SAD for the current frame, the combined  is modifiedas active sound and  decreases by 1.

#### 5.1.12.8 DTX hangover addition

For better DTX performance a version  of the combined  is generated through the addition of hangover. In this case there are two concurrent hangover logics that can extend the  active period. One is for DTX in general and one specifically to add additional DTX hangover in the case of music.

During the SAD initialization period the following variables are set as follows

 ()

The general DTX hangover works in the same way as the SAD1 hangover the main difference is in the hangover length. Also here the initial DTX hangover length depends on , initially the hangover is set to 2 frames and if the current input bandwidth is NB and  or it is set to 3, then follows a number of steps that may modify this start value. The modification depends on other input signal characteristics and codec mode.

The first two modifications increases the hangover length if there has already been a high activity, additional activity after a long burst has little effect on the total activity but can better cover short pauses. If there has been 12 or more active frames from the primary detector in SAD1 during the 16 last frames, that is , the allowed number of hangover frames is increased with 2 frames. Similarly if there has been 40 or more active frames for the final decision of SAD1 during the last 50 frames , that is , the allowed number of hangover frames is increased with 5 frames. At this point the allowed number of hangover frames may have been increased with 7 frames over the initial value, and to limit the total number of hangover frames it is therefore limited to . Another condition for limiting the hangover addition is if the primary activity becomes low there are different limits for different codec conditions, for AMR\_WB\_IO core the limit is 2, the same limit is also used for high SNR  for WB or SWB input in other conditions the limit is 3 frames. The condition for applying the limit is if the primary activity  or if the .

The DTX hangover can also be reduced if the final decision from SAD3 already includes a long hangover.

According to the noise type in SAD3, the decrement of the DTX hangover is set as shown in Table 14.

Table 14: Setting of the decrement of the DTX hangover

|  |  |  |
| --- | --- | --- |
| Bandwidth | Silence-type noise | Non-silence-type noise |
| NB | 0 | 1 |
| WB | 2 | 3 |
| SWB | 2 | 1 |

As for the hangover in SAD1 the counting of DTX hangover frames is reset only if at least 3 consecutive active speech frames () or if the SAD1 final decision has been active for over 45 of the 50 latest frames.

For the music hangover to start counting music hangover frames  AND  AND  AND  at which point the  for the next 15 frames or until hangover is terminated by the hangover termination logic, which can be triggered by the flag  which is described below.

The DTX hangover and the hangover described in subclause 5.1.12.3 when decisions from SAD1 and SAD2 are combined may be early terminated. The early hangover termination helps to increase the system capacity by saving unnecessary hangover frames. At each hangover frame, the comfort noise which will be produced at the decoder side is estimated at the encoder side, assuming if the current hangover frame would be encoded as the first SID frame after active burst. If the estimated comfort noise is found close to the noise characteristic maintained in the local CNG module in the encoder side, then no more hangover frame is considered needed and the hangover is terminated. Otherwise, hangover keeps on as long as the initial hangover length is not reached.

Specifically, the energy and the LSP spectrum of the comfort noise which will be produced at the decoder side are estimated at the encoder side. The energy of the current frame excitation is calculated

 ()

which is then converted to log domain

 ()

where  is the LP excitation of the current frame calculated in subclause 5.6.2.1.5,  is the frame length, is limited to non-negative value. An age weighted average energy, , is calculated from hangover frames except the current frame in the same way as described in sub-clause 6.7.2.1.2. The , together with the energy of the current frame excitation  are used to compute the estimated excitation energy for the comfort noise, .

 ()

where  is a smoothing factor, = 0.8 if , the number of hangover frames used for  calculation is less than 3, otherwise, = 0.95. The estimated excitation energy for the comfort noise is then converted to log domain

 ()

where  is bounded to non-negative value. An average LSP vector, , is calculated over the same hangover frames where the age weighted average energy  is calculated in the same way as described in sub-clause 6.7.2.1.2. The , together with the end-frame LSP vector of the current frame are used to compute the estimated LSP vector for the comfort noise, .

 ()

A set of energy and LSP difference parameters are calculated. The difference between the current frame log excitation energy and the log hangover average excitation energy is calculated.

 ()

The difference between the current frame end-frame LSP vector and the hangover average LSP vector is calculated.

 ()

where  is the order of LP filter. The difference between the estimated log excitation energy for the comfort noise and the current log excitation energy for the comfort noise kept in the local CNG module is calculated.

 ()

where  is the comfort noise excitation energy kept in the local CNG module as calculated in subclause 5.6.2.1.6. The difference between the estimated LSP vector for the comfort noise and the current LSP vector for the comfort noise kept in the local CNG module is calculated.

 ()

where  is the comfort noise LSP vector kept in the local CNG module as calculated in subclause 5.6.2.1.4. The maximum difference per LSP element between the estimated LSP vector for the comfort noise and the current LSP vector for the comfort noise kept in the local CNG module is calculated.

 ()

The hangover termination flag  is set to 1 if  and  and  and  and  when operating in VBR mode, or if  and  and  and  and  when operating in non-VBR mode. Otherwise  is set to 0. A  set to 1 means the current frame can be encoded as a SID frame even it is still in the hangover period. For safety reason of prevent CNG on short pauses between speech utterances, the actual encoding of SID frame is delayed by one frame.

### 5.1.13 Coding mode determination

To get the maximum encoding performance, the LP-based core uses a signal classification algorithm with six distinct coding modes tailored for each class of signal, namely the Unvoiced Coding (UC) mode, Voiced Coding (VC) mode, Transition Coding (TC) mode, Audio Coding (AC) mode, Inactive Coding (IC) mode and Generic Coding (GC) mode. The signal classification algorithm uses several parameters, some of them being optimized separately for NB and WB inputs.

Figure 13 shows a simplified high-level diagram of the signal classification procedure. In the first step, the SAD decision is queried whether the current frame is active or inactive. In case of inactive frame, IC mode is selected and the procedure is terminated. In the IC mode the inactive signal is encoded either in the transform domain by means of the AVQ technology or in the time/transform domain by means of the GSC technology, described below. In case of active frames, the speech/music classification algorithm is run to decide whether the current frame shall be coded with the AC mode. The AC mode, has been specifically designed to efficiently encode generic audio signals, particularly music. It uses a hybrid encoding technique, called the Generic Signal audio Coder (GSC) which combines both, LP-based coder operated in the time domain and a transform-domain coder. If the frame is not classified as “audio”, the classification algorithm continues with selecting unvoiced frames to be encoded with the UC mode. The UC mode is designed to encode unvoiced frames. In the UC mode, the adaptive codebook is not used and the excitation is composed of two vectors selected from a linear Gaussian codebook.

If the frame is not classified as unvoiced, then detection of stable voiced frames is applied. Quasi-periodic segments are encoded with the VC mode. VC selection is conditioned by a smooth pitch evolution. It uses ACELP technology, but given that the pitch evolution is smooth throughout the frame, more bits are assigned to the algebraic codebook than in the GC mode.

The TC mode has been designed to enhance the codec performance in the presence of frame erasures by limiting the usage of past information [19]. To minimize at the same time its impact on a clean channel performance, it is used only on the most critical frames from a frame erasure point of view – these are voiced frames following voiced onsets.

If a frame is not classified in one of the above coding modes, it is likely to contain a non-stationary speech segment and is encoded using a generic ACELP model (GC).



Figure 13: High-level diagram of the coding mode determination procedure

The selection of the coding modes is not uniform across the bitrates and input signal bandwidth. These differences will be described in detail in the subsequent sections. The classification algorithm starts with setting the current mode to GC.

#### 5.1.13.1 Unvoiced signal classification

The unvoiced parts of the signal are characterized by a missing periodic component. The classification of unvoiced frames exploits the following parameters:

– voicing measures

– spectral tilt measures

– sudden energy increase from a low level to detect plosives

– total frame energy difference

– energy decrease after spike

##### 5.1.13.1.1 Voicing measure

The normalized correlation, used to determine the voicing measure, is computed as part of the OL pitch searching module described in clause 5.1.10. The average normalized correlation is then calculated as

 (293)

where is defined in subclause 5.1.11.3.2.

##### 5.1.13.1.2 Spectral tilt

The spectral tilt parameter contains information about frequency distribution of energy. The spectral tilt is estimated in the frequency domain as a ratio between the energy concentrated in low frequencies and the energy concentrated in high frequencies, and is computed twice per frame.

The energy in high frequencies is computed as the average of the energies in the last two critical bands

 (294)

where  are the critical band energies, computed in subclause 5.1.5.2 and is the maximum useful critical band (= 19 for WB inputs and = 16 for NB inputs).

The energy in low frequencies is computed as the average of the energies in the first 10 critical bands for WB signals and in 9 critical bands for NB signals. The middle critical bands have been excluded from the computation to improve the discrimination between frames with high-energy concentration in low frequencies (generally voiced) and with high-energy concentration in high frequencies (generally unvoiced). In between, the energy content is not informative for any of the classes and increases the decision uncertainty.

The energy in low frequencies is computed differently for voiced half-frames with short pitch periods and for other inputs. For voiced female speech segments, the harmonic structure of the spectrum is exploited to increase the voiced-unvoiced discrimination. These frame are characterized by the following the condition

 (295)

where  are computed as defined in subclause 5.1.10.4 and the noise correction factor  as defined in subclause 5.1.10.6. For these frames,is computed bin-wise and only frequency bins sufficiently close to the speech harmonics are taken into account in the summation. That is

 ()

where  is the first bin (= 1 for WB inputs and = 3 for NB inputs) and  are the bin energies, as defined in subclause 5.1.5.2, in the first 25 frequency bins (the DC component is not considered). Note that these 25 bins correspond to the first 10 critical bands and that the first 2 bins not included in the case of NB input constitute the first critical band. In the summation above, only the terms related to the bins close to the pitch harmonics are considered. So is set to 1, if the distance between the nearest harmonics is not larger than a certain frequency threshold (50 Hz) and is set to 0 otherwise. The counter  is the number of the non-zero terms in the summation. In other words, only bins closer than 50 Hz to the nearest harmonics are taken into account. Thus, only high-energy terms will be included in the sum if the structure is harmonic at low frequencies. On the other hand, if the structure is not harmonic, the selection of the terms will be random and the sum will be smaller. Thus, even unvoiced sounds with high energy content in low frequencies can be detected. This processing cannot be done for longer pitch periods, as the frequency resolution is not sufficient. For frames not satisfying the condition (295), the low frequency energy is computed per critical band as

 (297)

for WB and NB inputs, respectively. The resulting low- and high-frequency energies are obtained by subtracting the estimated noise energy from the valuesand calculated above. That is

 ()

 ()

where  is the average noise energy in critical bands 18 and 19 for WB inputs, and 15 and 16 for NB inputs andis the average noise energy in the first 10 critical bands for WB input and in the critical bands 1-9 for NB inputs. They are computed similarly as in the two equations above. The estimated noise energies have been integrated into the tilt computation to account for the presence of background noise.

Finally, the spectral tilt is given by

 ()

For NB signals, the missing bands are compensated by multiplying  by 6. Note that the spectral tilt computation is performed twice per frame to obtainand, corresponding to both spectral analyses per frame. The average spectral tilt used in unvoiced frame classification is given by

 ()

where is the tilt in the second half of the previous frame.

##### 5.1.13.1.3 Sudden energy increase from a low energy level

The maximum energy increase  from a low signal level is evaluated on eight short-time segments having the length of 32 samples. The energy increase is then computed as the ratio of two consecutive segments provided that the first segment energy was sufficiently low. For better resolution of the energy analysis, a second pass is computed where the segmentation is done with a 16 sample offset. The short-time maximum energies are computed as

 (302)

where corresponds to the last segment of the previous frame, and corresponds to the current frame. The second set of maximum energies is computed by shifting the speech indices in equation () by 16 samples. That is

 ()

 ()

The maximum energy variation is computed as follows:

 ()

##### 5.1.13.1.4 Total frame energy difference

The classification of unvoiced frames is further improved by taking into account the difference of total frame energy. This difference is calculated as



where  is the total frame energy calculated in subclause 5.1.5.2 and  is the total frame energy in the previous frame.

##### 5.1.13.1.5 Energy decrease after spike

The detection of energy decrease after a spike prevents the UC mode on significant temporal events followed by relatively rapid energy decay. Typical examples of such signals are castanets.

The detection of the energy decrease is triggered by detecting a sudden energy increase from a low level as described in subclause 5.1.13.1.3. The maximum energy variation  must be higher than 30 dB. Further, for NB inputs the mean correlation must not be too high, i.e. the condition  must be satisfied too.

The energy decrease after a spike is searched within 10 overlapped short-time segments (of both sets of energies) following the detected maximum energy variation. Let’s call  the index for which  was found, and the corresponding set of energies . If , then the searched interval is for both sets. If , then the searched interval is for the 2nd set, but  for the 1st set of energies.

The energy decrease after a spike is then searched as follows. As the energy can further increase beyond the segment  for which  was found, the energy increase is tracked beyond that segment to find the last segment with energy still monotonically increasing. Let’s denote the energy of that segment . Starting from that segment until the end of the searched interval, the minimum energy  is then determined. The detection of an energy decrease after spike is based on the ratio of the maximum and minimum energies

 ()

This ratio is then compared to a threshold of 21 dB for NB inputs and 30 dB for other inputs.

The detection of energy decrease after a spike further uses a hysteresis in the sense that UC is prevented not only in the frame where is above the threshold (), but also in the next frame (). In subsequent frames (), the hysteresis is reset () only if the following condition is met:

. ()

Given that the searched interval of overlapped segments is always 10, it can happen that the detection cannot be completed in the current frame if a rapid energy increase happens towards the frame end. In that case, the detection is completed in the next frame, however, as far as the hysteresis logic is concerned, the detection of energy decrease after a spike still pertains to the current frame.

##### 5.1.13.1.6 Decision about UC mode

To classify frames for encoding with UC mode, several conditions need to be met. As the UC mode does not use the adaptive codebook and no long-term prediction is thus exploited, it is necessary to make sure that only frames without periodic content are coded with this mode. The decision logic is somewhat different for WB and NB inputs and is described for both cases separately.

For WB inputs, all of the following conditions need to be satisfied to select the UC mode for the current frame.

1. Normalized correlation is low:



1. Energy is concentrated in high frequencies.



1. The current frame is not in a segment following voiced offset:



where  is the raw coding mode selected in the previous frame (described later in this document).

1. There is no sudden energy increase:



1. The current frame is not in a decaying segment following sharp energy spike:



For NB inputs, the following conditions need to be satisfied to classify the frame for NB UC coding.

1. Normalized correlation is low:



1. Energy is concentrated in high frequencies.



1. The current frame is not in a segment following voiced offset:



where  is the raw coding mode selected in the previous frame (described later in this document).

1. There is no sudden energy increase:



1. The current frame is not in a decaying segment following sharp energy spike:



#### 5.1.13.2 Stable voiced signal classification

The second step in the signal classification algorithm is the selection of stable voiced frames, i.e. frames with high periodicity and smooth pitch contour. The classification is mainly based on the results of the fractional open-loop pitch search described in section 5.1.10.9. As the fractional open-loop pitch search is done in a similar way as the closed-loop pitch search, it is assumed that if the open-loop search gives a smooth pitch contour within predefined limits, the optimal closed-loop pitch search would give similar results and limited quantization range can then be used. The frames are classified into the VC mode if the fractional open-loop pitch analysis yields a smooth contour of pitch evolution over all four subframes. The pitch smoothness condition is satisfied if  , for i = 0, 1, 2, where  is the fractional open-loop pitch lag found in subframe *i* (see section 5.1.10.9 for more details). Furthermore, in order to select VC mode for the current frame the maximum normalized correlation  must be greater than 0.605 in each of the four subframes. Finally, the spectral tilt  must be higher than 4.0.

The decision about VC mode is further improved for frames with stable short pitch evolution and high correlation (e.g. female or child voices or opera voices). Pitch smoothness is again satisfied if , for i = 0, 1, 2. High correlation is achieved in frames for which the mean value of the normalized correlation in all four subframes is higher than 0.95 and the mean value of the smoothed normalized correlation is higher than 0.97. That is

 ()

The smoothing of the normalized correlation is done as follows

 ()

Finally, VC mode is also selected in frames for which the flag  = *Stab\_short\_pitch\_flag = flag\_spitch* has been previously set to 1 in the module described in sub-clause 5.1.10.8. Further, when the signal has very high pitch correlation,  is also set to 1 so that the VC mode is maintained to avoid selecting Audio Coding (AC) mode later, as follows,

If (=1 or

(dpit1 <= 3 AND dpit2 <= 3 AND dpit3 <= 3 AND  > 0.95 AND  > 0.97))

{

VC = 1;

=1

}

wherein *, , ,* and are defined in subclause 5.1.10.8.

The decision taken so far (i.e. after UC and VC mode selection) is called the “raw” coding mode, denoted. The value of this variable from the current frame and from the previous frame is used in other parts of the codec.

#### 5.1.13.3 Signal classification for FEC

This subclause describes the refinement of the signal classification algorithm described in the previous section in order to improve the codec's performance for noisy channels. The classification used to select UC and VC frames cannot be directly used in the FEC as the purpose of the classification is not the same. Instead, better performance could be achieved by tuning both classification aspects separately.

The basic idea behind using a different signal classification approach for FEC is the fact that the ideal concealment strategy is different for quasi-stationary speech segments and for speech segments with rapidly changing characteristics. Whereas the best processing of erased frames in non-stationary speech segments can be summarized as a rapid drop of energy, in the case of quasi-stationary signal, the speech-encoding parameters do not vary dramatically and can be kept practically unchanged during several adjacent erased frames before being damped. Also, the optimal method for a signal recovery following an erased block of frames varies with the classification of the speech signal.

Furthermore, this special classification information is also used to select frames to be encoded with the TC mode (see subclause 5.1.13.4).

To distinguish the signal classification algorithm for FEC from the signal classification algorithm for coding mode determination (described earlier in subclauses 5.1.13.1 and 5.1.13.2), we will refer here to “signal class” rather than “coding mode” and denote it .

##### 5.1.13.3.1 Signal classes for FEC

The frame classification is done with the consideration of the concealment and recovery strategy in mind. In other words, any frame is classified in such a way that the concealment can be optimal if the following frame is missing, and that the recovery can be optimal if the previous frame was lost. Some of the classes used in the FEC do not need to be transmitted, as they can be deduced without ambiguity at the decoder. Here, five distinct classes are used and defined as follows:

• INACTIVE CLASS comprises all inactive frames. Note, that this class is used only in the decoder.

• UNVOICED CLASS comprises all unvoiced speech frames and all frames without active speech. A voiced offset frame can also be classified as UNVOICED CLASS if its end tends to be unvoiced and the concealment designed for unvoiced frames can be used for the following frame in case it is lost.

• UNVOICED TRANSITION CLASS comprises unvoiced frames with a possible voiced onset at the end. The onset is however still too short or not built well enough to use the concealment designed for voiced frames. The UNVOICED TRANSITION CLASS can only follow a frame classified as UNVOICED CLASS or UNVOICED TRANSITION CLASS.

• VOICED TRANSITION CLASS comprises voiced frames with relatively weak voiced characteristics. Those are typically voiced frames with rapidly changing characteristics (transitions between vowels) or voiced offsets lasting the whole frame. The VOICED TRANSITION CLASS can only follow a frame classified as VOICED TRANSITION CLASS, VOICED CLASS or ONSET CLASS.

• VOICED CLASS comprises voiced frames with stable characteristics. This class can only follow a frame classified as VOICED TRANSITION CLASS, VOICED CLASS or ONSET CLASS.

• ONSET CLASS comprises all voiced frames with stable characteristics following a frame classified as UNVOICED CLASS or UNVOICED TRANSITION CLASS. Frames classified as ONSET CLASS correspond to voiced onset frames where the onset is already sufficiently built for the use of the concealment designed for lost voiced frames. The concealment techniques used for frame erasures following the ONSET CLASS are the same as those following the VOICED CLASS. The difference is in the recovery strategy.

• AUDIO CLASS comprises all frames with harmonic or tonal content, especially music. Note that this class is used only in the decoder.

##### 5.1.13.3.2 Signal classification parameters

The following parameters are used for the classification at the encoder: normalized correlation, , spectral tilt measure, , pitch stability counter, *pc*, relative frame energy, , and zero crossing counter, *zc*. The computation of these parameters which are used to classify the signal is explained below.

The normalized correlation, used to determine the voicing measure, is computed as part of the OL pitch analysis module described in subclause 5.1.10. The average correlation is defined as

 ()

where and  are the normalized correlation of the second half-frame and the look ahead, respectively.

The spectral tilt measure, , is computed as the average (in dB) of both frame tilt estimates, as described in subclause 5.1.13.1.2. That is

 ()

The pitch stability counter, *pc*, assesses the variation of the pitch period. It is computed as follows:

 ()

where the values ,  and correspond to the three OL pitch estimates evaluated in each frame (see subclause 5.1.10).

The last parameter is the zero-crossing rate, *zc*, computed on the second half of the current speech frame and the look-ahead. Here, the zero-crossing counter, *zc*, counts the number of times the signal sign changes from positive to negative during that interval. The zero-crossing rate is calculated as follows

 ()

where the function sign[.] returns +1 if the value is positive or -1 it is negative.

##### 5.1.13.3.3 Classification procedure

The classification parameters are used to define a function of merit, . For that purpose, the classification parameters are first scaled between 0 and 1 so that each parameter translates to 0 for an unvoiced signal and to 1 for a voiced signal. Each parameter, , is scaled by a linear function as follows:

 ()

and clipped between 0 and 1 (except for the relative energy which is clipped between 0.5 and 1). The function coefficients,  and , have been found experimentally for each of the parameters so that the signal distortion due to the concealment and recovery techniques used in the presence of frame erasures is minimal. The function coefficients used in the scaling process are summarized in the following table.

Table 15: Coefficients of the scaling function for FEC signal classification

|  |  |  |  |
| --- | --- | --- | --- |
| parameter | description |  |  |
|  | normalized correlation | 2.857 | -1.286 |
|  | spectral tilt | 0.04167 | 0 |
|  | pitch stability counter | -0.07143 | 1.857 |
|  | relative frame energy | 0.05 | 0.45 |
|  | zero-crossing counter | -0.04 | 2.4 |

The function of merit has been defined as

 ()

where the superscript *s* indicates the scaled version of the parameters. The classification is then done using the function of merit, , and following the rules summarized in the following table.

Table : Rules for FEC signal classification

|  |  |  |
| --- | --- | --- |
| previous class | rule | selected class |
| VOICED CLASS  ONSET CLASS  VOICED TRANSITION CLASS |  | VOICED CLASS |
|  | VOICED TRANSITION CLASS |
|  | UNVOICED CLASS |
| UNVOICED CLASS  UNVOICED TRANSITION CLASS |  | ONSET CLASS |
|  | UNVOICED TRANSITION CLASS |
|  | UNVOICED CLASS |

For the purpose of FEC signal classification, all inactive speech frames, unvoiced speech frames and frames with very low energy are directly classified as UNVOICED CLASS. This is done by checking the following condition

 ()

which has precedence over the rules defined in the above table.

#### 5.1.13.4 Transient signal classification

As a compromise between the clean-channel performance of the codec and its robustness to channel errors, the use of the TC mode is limited only to a single frame following voiced onsets and to transitions between two different voiced segments. Voiced onsets and transitions are the most problematic parts from the frame erasure point of view. Therefore, the frame after the voiced onset and voiced transitions must be as robust as possible. If the transition/onset frame is lost, the following frame is encoded using the TC mode, without the use of the past excitation signal, and the error propagation is broken.

The TC mode is selected according to the counter of frames from the last detected onset/transition . The onset/transition detection logic is described by the state machine in the following diagram.



Figure 14 : TC onset/transition state machine

In the above logic,  is set to 0 for all inactive frames, resp. frames for which the FEC signal class is either UNVOICED CLASS or UNVOICED TRANSITION CLASS. When the first onset frame is encountered  is set to 1. This is again determined by the FEC signal class. The onset/transition frame is always coded with the GC mode, i.e. the coding mode is set to GC in this frame. The next active frame after the onset frame increases  to 2 which means that there is a transition and TC mode is selected. The counter is set to -1 if none of the above situations happens waiting for the next inactive frame. Naturally, the GC/TC mode is selected by the above logic only when the current frame is an active frame, i.e. when .

#### 5.1.13.5 Modification of coding mode in special cases

In some special situations, the decision about coding mode is further modified. This is e.g. due to unsuitability of the selected coding mode at particular bitrate or due to signal characteristics which make the selected mode inappropriate. For example, the UC mode is supported only up to 9.6 kbps after which it is replaced by the GC mode. The reason is that for bitrates higher than 9.6 kbps the GC mode already has enough bits to fully represent the random content of an unvoiced signal. The UC mode is also replaced by the GC mode at 9.6 kbps if the counter of previous AC frames is bigger than 0. The counter of AC frames is initialized to the value of 200, incremented by 10 in every AC frame and decremented by 1 in other frames but not in IC frames. The counter is upper limited by 1000 and lower limited by 0.

At 32 and 64 kbps, only GC and TC modes are employed, i.e. if the coding mode has been previously set to UC or VC, it is overwritten to GC. Finally, for certain low-level signals, it could happen that the gain quantizer in the NB VC mode goes out of its dynamic range. Therefore, the coding mode is changed to GC mode if the relative frame energy dB but only at 8.0 kbps and lower bitrates.

The coding mode is also changed to the TC mode in case of mode switching. If, in the previous frame, 16 kHz ACELP core was used but the current frame uses 12.8 kHz ACELP core, it is better to prevent potential switching artefacts resulting from signal discontinuity and incorrect memory. This modification is done only for frames other than VC, i.e. if , where  is the raw coding mode described in subclause 5.1.13.2. Further, the modification takes place only for active frames in case of DTX operation.

The coding mode is overridden to the TC mode if MDCT-based core was used in the last frame but the current frame is encoded with an LP-based core. Finally, the coding mode is changed to the TC mode if the EVS codec is operated in the DTX mode and if the last frame was a SID frame encoded with the FD\_CNG technology.

#### 5.1.13.6 Speech/music classification

Music signals, in general, are more complex than speech and conform less to any known LP-based model. Therefore, it is of interest to distinguish music signals (generic audio signals) from speech signals. Speech/music classification then allows using a different coding approach to such signals. This new approach has been called the Generic audio Signal Coding mode (GSC), or the Audio Coding (AC) mode.

The speech/music classification is done in two stages. The first stage of the speech/music classifier is based on the Gaussian Mixture Model (GMM) and performs the best statistically based discrimination of speech from generic audio. The second stage has been optimized directly for the GSC mode. In other words, the classification in the second stage is done in such a way that the selected frames are suitable for the AC mode. Each speech/music classifier stage yields its own binary decision,  and  which is either 1 (music) or 0 (speech or background noise). The speech decision and the background noise decision have been grouped together only for the purposes of the speech/music classification. The selection of the IC mode for inactive signals incl. background noise is done later in the codec and described in subclause 5.1.13.5.7.

The decisions of the first and the second stage are refined and corrected for some specific cases in the subsequent modules, described below. The final decision about the AC mode is done based on  and  but the two flags are also used for the selection of the coder technology which is described in subclause 5.1.16.

##### 5.1.13.6.1 First stage of the speech/music classifier

The GMM model has been trained on a large database of speech and music signals covering several male and female speakers, multiple languages and various genres of instrumental and vocal music. The statistical model uses a vector of 12 unique features, all normalized to a unit interval and derived from the basic parameters that have been calculated in the pre-processing part of the encoder. There are three statistical models inside the GMM: speech, music and noise. The statistical model of the background noise has been added to improve the SAD algorithm described in subclause 5.1.12. Each statistical model is represented by a mixture of six normal (Gaussian) distributions, determined by their relative weight, mean and full covariance matrix. The speech/music classifier exploits the following characteristics of the input signal:

– OL pitch

– normalized correlation

– spectral envelope (LSPs)

– tonal stability

– signal non-stationarity

– LP residual error

– spectral difference

– spectral stationarity



Figure 15 : Schematic diagram of the first stage of the speech/music classifier

The OL pitch feature is calculated as the average of the three OL pitch estimates, i.e.

 ()

where are computed as in subclause 5.1.10.7. In onset/transition frames and in the TC frame after, it is better to use only the OL pitch estimate of the second analysis window, i.e. .

The normalized correlation feature  used by the speech/music classifier is the same one as used in the unvoiced signal classification. See subclause 5.1.13.1.1. for the details of its computation. In onset/transition frames and in the TC frame after, it is better to use only the correlation value of the second analysis window, i.e. .

There are five LSF parameters used as features inside the first stage of the speech/music classifier. These are calculated as follows

 ()

Another feature used by the speech/music classifier is the correlation map which is calculated as part of the tonal stability measure in subclause 5.1.11.2.5. However, for the purposes of speech/music classification, it is not the long-term correlation map which is summed but rather the correlation map of the current frame. The reason is to limit the impact of past information on the speech/music decision in the current frame. That is

 ()

In case of NB signals, the value of  is multiplied by 1.53.

Signal non-stationarity is also used in the speech/music classifier but its calculation is slightly different than in the case of background noise estimation described in subclause 5.1.11.2.1. Firstly, the current log-energy per band is defined as

, *i*=2,.,16 ()

Then,

 ()

The LP residual log-energy ratio is calculated as

 ()

where the superscript [-1] denotes values from the previous frame. In case of NB signals, the statistical distribution of  is significantly different than in case of WB signals. Thus, for NB signals .

For the last two features, the power spectrum must be normalized as follows:

, *k*=3,..,69 ()

and difference spectrum calculated as follows:

, *k*=3,..,69 ()

Then we calculate the spectral difference as the sum of  in the log domain. That is

 ()

The spectral non-stationarity is calculated as the product of ratios between power spectrum and the difference spectrum. This is done as follows

 ()

##### 5.1.13.6.2 Scaling of features in the first stage of the speech/music classifier

The feature vector *FVi, i*=0,..,11 is scaled in such a way that all its values are approximately in the range [0;1]. This is done as

 *i*=0,..,11 (327)

where the scaling factors *sfai* and *sfbi* have been found on a large training database. The scaling factors are defined in the following table.

Table : Scaling factors for feature vector in the speech/music classifier

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *i* | WB | | NB | |
|  |  |  |  |
| 0 | 0.048 | -0.0952 | 0.0041 | 0 |
| 1 | 1.0002 | 0 | 0.8572 | 0.1020 |
| 2 | 0.6226 | -0.0695 | 0.6739 | -0.1000 |
| 3 | 0.5497 | -0.1265 | 0.6257 | -0.1678 |
| 4 | 0.4963 | -0.2230 | 0.5495 | -0.2380 |
| 5 | 0.5049 | -0.4103 | 0.5793 | -0.4646 |
| 6 | 0.5069 | -0.5717 | 0.2502 | 0 |
| 7 | 0.0041 | 0 | 0.0041 | 0 |
| 8 | 0.0022 | -0.0029 | 0.0020 | 0 |
| 9 | 0.0630 | 1.0015f | 0.0630 | 1.0015 |
| 10 | 0.0684 | 0.9103f | 0.0598 | 0.8967 |
| 11 | 0.1159 | -0.2931 | 0.0631 | 0 |

##### 5.1.13.6.3 Log-probability and decision smoothing

The multivariate Gaussian probability distribution is defined as

 ()

where **FV** is the feature vector, **µ** is the vector of means and **Σ** is the variance matrix. As stated before, the dimension of the feature vector is *k*=12. The means and the variance matrix are found by the training process of the Gaussian Mixture Model (GMM). This training is done by means of the EM (Expectation-Maximization) algorithm . The speech/music classifier is trained with a mixture of 6 Gaussian distributions. The log-likelihood of each mixture is defined as

 *i*=1,..,6 ()

where *wi* is the weight of each mixture. The term  is calculated in advance and stored in the form of a look-up table. The probability over the complete set of 6 Gaussians is then calculated in the following way:

 ()

and the log-likelihood over the complete set as

 ()

Since there are three trained classes in the GMM model (speech, music and noise), three log-likelihood values are obtained by the above estimation process: . Only  and  are used in the subsequent logic.  is used in the SAD algorithm to improve detection accuracy. Therefore, in case of inactive signal when *fLSAD\_HE* is 0

 ()

The difference of log-likelihood is calculated as

 ()

which can be directly interpreted as a speech/music decision without hangover. This decision has low dynamic range and fluctuates a lot around zero, especially for mixed signals. To improve the detection accuracy, the decision is smoothed by means of AR filtering

 ()

where the superscript [-1] denotes the previous frame and  is the filtering factor which is adaptively set on a frame-by-frame basis. The filtering factor is in the range [0;1] and is based on two measures (weighting factors). The first weighting factor  is related to the relative frame energy and the second weighting factor  is designed to emphasize rapid negative changes of .

The energy-based weight  is calculated as follows

 ()

where  is relative frame energy. The result of the addition means that the weight has values close to 0.01 in low-energy segments and close to 1 in high energy, or more important, segments. Therefore, the smoothed decision follows the current decision more closely if the signal energy is relatively high and leads to past information being disregarded more readily. On the other hand, if the signal energy is low, the smoothed decision puts more emphasis on previous decisions rather than the current one. This logic is motivated by the observation that discrimination between speech and music is more difficult when the SNR of the signal is low.

The second weighting factor  is designed to track sudden transitions from music to speech. This situation happens only in frames where  and at the same time . In these frames

 ()

where the parameter  is a quantitative measure of sudden falls, or drops, in the value of . This parameter is set to 0 in all frames that do not fulfil the previous condition. In the first frame when  falls below 0, it is set equal to its negative value and it is incremented when  continues to fall in consecutive frames. In the first frame when  stops decreasing it is reset back to 0. Thus,  is positive only during frames of falling  and the bigger the fall the bigger its value. The weighting factor  is then calculated as

 ()

The filtering factor is then calculated by from the product of both weights, i.e.

 ()

##### 5.1.13.6.4 State machine and final speech/music decision

The state machine is an event-based decision system in which the state of the speech/music classifier is changed from INACTIVE to ACTIVE and vice-versa. There are two intermediate states: ENTRY and INSTABLE. The classifier must always go through the ENTRY state in order to be ACTIVE. During the ENTRY period, there is no relevant past information that could be exploited by the algorithm for hangover additional described later in this section. When the classifier is in the ACTIVE state but the energy of the input signal goes down up to the point when it is almost equal to the estimated background noise energy level, the classifier is in an UNSTABLE state. Finally, when SAD goes to 0, the classifier is in INACTIVE state. The following state-flow diagram shows the transitions between the states.



Figure 16 : State machine for the first stage of the speech/music classifier

The conditions for changing the states are described in from of decision tree in figure . The processing starts at the top-left corner and stops at bottom-right corner. The counter of inactive states  is initialized to 0 and the state variable  is initialized to -8. The state variable stays within the range [-8;+8] where the value of -8 means INACTIVE state and the value of +8 means ACTIVE STATE. If  the classifier is in ENTRY state and if  the classifier is in INSTABLE state.

If the speech/music classifier is in INACTIVE state, i.e. if  then the smoothed decision is automatically set to 0, i.e. .

The final decision of the speech/music classifier is binary and it is characterized by the flag *fSM*. The flag is set to 0 if  and the classifier is in INACTIVE STATE, i.e. when . If there is a transition from the ACTIVE state to the INACTIVE or INSTABLE state, characterized by , the flag retains its value from the previous frame. If the classifier is in ENTRY state, characterized by , the flag is set according to weighted average of past non-binary decisions. This is done as follows

 ()

where the weighting coefficients  are given in the following table:

Table : Weighting coefficients for the ENTRY period of the speech/music classifier

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 2 | 0.6 | 0.4 | 0 | 0 | 0 | 0 | 0 | 0 |
| 3 | 0.47 | 0.33 | 0.2 | 0 | 0 | 0 | 0 | 0 |
| 4 | 0.4 | 0.3 | 0.2 | 0.1 | 0 | 0 | 0 | 0 |
| 5 | 0.3 | 0.25 | 0.2 | 0.15 | 0.1 | 0 | 0 | 0 |
| 6 | 0.233 | 0.207 | 0.18 | 0.153 | 0.127 | 0.1 | 0 | 0 |
| 7 | 0.235 | 0.205 | 0.174 | 0.143 | 0.112 | 0.081 | 0.05 | 0 |
| 8 | 0.2 | 0.179 | 0.157 | 0.136 | 0.114 | 0.093 | 0.071 | 0.05 |



Figure 17 : Decision tree for transitions between INACTIVE and ACTIVE states of the speech/music classifier

The flag  is set to 1 when  > 2. If the classifier is in a stable ACTIVE state, the flag retains its value from the previous frame unless one of the following two situations happens. If  but the decisions in the previous three frames were all “speech”, i.e. for *i*=-1,-2,-3, there is a transition from speech to music and  is set to 1. If  but the decision in the previous frame was “music”, i.e. , there is a transition from music to speech and  is set to 0.

The speech/music decision obtained by the algorithm described so far will be denoted  in the following text to distinguish it from the second stage of the speech/music classifier in which the decision will be denoted .

##### 5.1.13.6.5 Improvement of the classification for mixed and music content

The speech/music decision  obtained above is further refined with the goal of improving the classification rate on music and mixed content. A set of feature parameters are extracted from the input signal and buffered. Statistical analysis is performed on each feature parameter buffer and a binary speech/music decision  is obtained using a tree-based classification. During the processing the value ‘1’ indicates music and the value ‘0’ indicates non-music. As a result of this refinement, the earlier speech/music decision may be adjusted from ‘0’ to ‘1’ if  has a final value of ‘1’ in the situation that the and  are not aligned with each other.

The feature parameters used to form the feature parameter buffers include a spectral energy fluctuation parameter,, a tilt parameter of the LP analysis residual energies, , a high-band spectral peakiness parameter, , a parameter of correlation map sum, , a voicing parameter, , and finally three tonal parameters , and . Since music is assumed to only existing during high SNR active regions of the input signal, the classification refinement is only applied for active frames and when the long-term SNR is high. So, if the SAD flag  indicates that the current frame is an inactive frame or the long-term SNR  is below a threshold of 25, i.e. , then the classification refinement is terminated without executing fully. In the early termination case, the speech/music decision is kept unchanged and two long-term speech/music decisions , , as will be described later in this subclause, are both initialized to 0.5.

Before computing the various feature parameters, percussive music is first detected. Percussive music is characterized by temporal spike-like signals. First the log maximum amplitude of the current frame is found as

 ()

where  is the time-domain input frame. The difference between the log maximum amplitude and its moving average from the previous frame is calculated

 ()

where the superscript [-1] denotes the value from the previous frame.  is updated at each frame after the calculation of , if both the normalized pitch correlations of the current frame  and  as calculated in defined in subclause 5.1.11.3.2 are greater than 0.9 as

 ()

where the value is the forgetting factor and is set to 0.75 for increasing updates () and 0.995 for decreasing updates (). The , the total frame energy  calculated in subclause 5.1.5.2, the normalized pitch correlation  defined in subclause 5.1.11.3.2 and the long-term active signal energy  are used to identify the temporal spike-like signals. First, certain energy relationship between several past frames is checked. If  and  and  and  and , where the superscript  denotes the -th frame in the past, the energy envelope of temporal spike-like signal is considered found. Then if the voicing is low, that is if, the percussive music flag  is set to 1 indicating the detection of spike-like signal, if the the normalized pitch correlations for the second half of the previous frame, the first half of the current frame and the second half of the current frame, ,and are all less than 0.75 and the  is greater than 10, or if simply the long-term speech/music decision  is greater than 0.8.

Besides the detection of percussive music, sound attacks are also detected using , ,  and , where  denotes the  of the previous frame. If  and  and  and , sound attack is detected and the attack flag  is set to 3. The attack flag  is decremeted by 1 in each frame afterthe calculation and buffering of the spectral energy fluctuation parameterwhich is claculated from the log energy spectrum of the current frame as follows: Firstly, all local peaks and valleys in the log spectrum , as calculated in equation (127) are identified. A value of  is considered as a local peak if  and . A value of  is considered as a local valley if  and . Besides, the first local valley is found as the  with and, the last local valley is found as the  with and. For each local peak, its peak to valley distance is calculated as

 ()

where  denotes the peak to valley distance of the -th local peak,  denotes the log energy of the -th local peak and ,  denote the respect log energy of the local valleys adjacent to the -th local peak at the lower frequency side and the higher frequency side,  denotes the number of local peaks. An array called peak to valley distance map is then obtained as

 ()

where  denotes the peak to valley distance map,  denotes the index (or the location) of the -th local peak in the log spectrum . The spectral energy fluctuation parameter  is defined as the average energy deviation between the current frame spectrum and the spectrum two frames ago at locations of the local spectral peaks . The  is computed as

 (345)

where  and  denote respectively the log energy spectrum of the current frame and the log energy spectrum of the frame two frames ago,  denotes the number of local peaks. If = 0,  is set to 5. The computed  is stored into a buffer  of 60 frames if there is no sound attack in the past 3 frames (including the current frame), that is if . Moreover, if the long-term speech/music decision is greater than 0.8 meaning a strong music signal in previous classifications, then the value of is upper limited to 20 before it is stored into the . The  buffer  is altered at every first active frame after an inactive segment (flagged by) that all values in the buffer excluding the one just calculated and stored for the current frame are changed to negative values.

The effective portion of the buffer  is determined in each frame after the calculation and buffering of the parameter. The effective portion is defined as the portion in the  buffer which contains continuous non-negative values starting from the value of the latest frame . If percussive music is detected, that is if the percussive music flag  is set to 1, each value in the effective portion of the  buffer  is initialized to 5.

The tilt parameter of the LP analysis residual energies is calculated as

 (346)

where  is the LP error energies computed by the Levinson-Durbin algorithm. The computed  is stored into a buffer  of 60 frames.

The high-band spectral peakiness parameter  reflects an overall tonality of the current frame at its higher frequency band and is calculated from the peak to valley distance map  as

 ()

The calculated  is stored into a buffer  of 60 frames.

The three tonal parameters , and  are also calculated from the peak to valley distance map . denotes the first number of harmonics found from the spectrum of the current frame.  is calculated as

 ()

 denotes the second number of harmonics also found from the spectrum of the current frame.  is defined more strictly than  and is calculated as

 ()

denotes the number of harmonics found only at the low frequency band of the current frame’s spectrum and is calculated as

 ()

The calculated values of , and  are stored into their respective buffers ,  and all of 60 frames.

The sum of correlation map  as calculated by

 ()

is also stored into a buffer  of 60 frames, where  is the correlation map calculated in subclause 5.1.11.2.5.

The voicing parameter  is defined as the difference of log-likelihood between speech class and music class as calculated in subclause 5.1.13.6.3. The  is calculated as

 ()

where ,  are the log-likelihood of speech class and the log-likelihood of music class respectively. is stored into a buffer  of 10 frames.

The speech/music decision  is obtained through a tree-based classification. The  is first initialized as a hysteresis of the long-term speech/music decision  from the previous frame, i.e.

 ()

where the superscript [-1] denotes the value from the previous frame. Then, the  can be altered through successive classifications. Let  denotes the length of the effective portion in. Depending on the actual value of , different classification procedures are followed. If , insufficient data is considered in the feature parameter buffers. The classification is terminated and the initialized  is used as the final . If , the respective mean values , and are calculated from ,  and  over the effective portion and the variance , calculated over the effective portion from  is also obtained. In addition, the number of positive values  among the 6 latest values in  is counted. The speech/music decision  is then set to 1 if and any of the following conditions is fulfilled;  or  or  or . Otherwise, if , the feature buffers are first analysed over the portion  containing the latest 10 values. The mean values , and  are calculated from, and  over and for the same portion the variance  is also calculated from .Besides, the mean value of, over a shorter portion of the latest 5 frames is also calculated. The  is found as the number of positive values in. The speech/music decision  is determined without the need to analyse any longer portion if strong speech or music characteristics are found within , that is, the  and  are both set to 1 if  and  and any of the following conditions is fulfilled:  or  or  or  . The  and  are both set to 0 if any of the following conditions is fulfilled:  or  or  or . If no class is determined for  over the values, the  is determined iteratively over portions starting from  until the whole effective portion is reached. For each iteration, the respective mean values, and  are calculated from,  and  over the portion under analysis and for the same portion the variance  is also calculated from . The mean value is calculated from  over , and the number of positive values in ,, is also counted. The value of  is set to 1 if  and any of the following conditions is fulfilled:  or  or  or. If through the above iteration procedure the  is not set and if the effective portion reaches the maximum of 60 frames, a final speech/music discrimination is made from ,  and . The mean value  of the , the sum value  of the , and the sum value  of the  are calculated over the whole buffers. A low frequency tonal ratio  is calculated as

 ()

The  is set to 1 if  or . Otherwise, if , the is set to 0.

If  is greater than 30, then both the two long-term speech/music decisions  and  are updated at each frame with  as

 (355)

 (356)

where the superscript [-1] denotes the value from the previous frame. If the total frame energy  calculated in subclause 5.1.5.2 is greater than 1.5 and  is less than 2 and the raw coding mode  is either UNVOICED or INACTIVE, then an unvoiced counter  initialized to 300 at the first frame is updated by

 ()

Otherwise,  is incremented by 1. The value of  is bounded between [0, 300]. The  is further smoothed by an AR filtering as

 ()

where is the smoothed , the superscript [-1] denotes the value from the previous frame. If  is set to 1 in any previous stage, the flag  is overridden by  unless the long-term speech/music decision  as calculated in equation (356) is close to speech and the smoothed unvoiced counter exhibits strong unvoiced characteristic, that is, the  is set to 1 if  and  .

##### 5.1.13.6.6 Second stage of the speech/music classifier

The second stage of the speech/music classifier has been designed and optimized for the GSC technology. Not all frames classified as music in the first stage can be encoded directly with the GSC technology due to its inherent limitations. Therefore, in the second stage of the speech/music classifier, a subset of frames that have been previously classified as “music”, i.e. for which , are reverted to speech and encoded with one of the CELP modes. The decision in the second stage of the speech/music classifier is denoted . The second stage is run only for WB, SWB and FB signals, not for NB. The reason for this limitation is purely due to the fact that GSC technology is only applied at bandwidths higher than NB.

The second stage of the speech/music classifier starts with signal stability estimation which is based on frame-to-frame difference of the total energy of the input signal. That is

 ()

Then, the mean energy difference is calculated as

 ()

i.e. over the period of the last 40 frames. Then, the statistical deviation of the delta-energy values around this mean is calculated as

 ()

i.e. over the period of the last 15 frames.

After signal stability estimation, correlation variance is calculated as follows. First, mean correlation is estimated over the period of the last 10 frames. This is done as

 ()

where  and the superscript [-1] is used to denote past frames. Then, the correlation variance is defined as

 ()

In order to discriminate highly-correlated stable frames, long-term correlation is calculated as

 ()

The flag  is set to 1 if  and at the same time .

In the next step, attacks are detected in the inputs signal. This is done by dividing the current frame of the input signal into 32 segments where each segment has the length of 8 samples. Then, energy is calculated in each segment as

 ()

The segment with the maximum energy is then found by

 ()

and this is the position of the candidate attack. In all active frames where  and for which the coding mode was set to GC, the following logic is executed to eliminate false attacks, i.e. attacks that are not sufficiently strong. First, the mean energy in the first 3 sub-frames is calculated as

 ()

and the mean energy after the detected candidate attack is defined

 ()

and the ratio of these two energies is compared to a certain threshold. That is

 ()

Thus, the candidate attack position is set to 0 if the attack is not sufficiently strong. Further, if the FEC class of the last frame was VOICED CLASS and if  then  is also set to 0.

To further reduce the number of falsely detected attacks, the segment with maximum energy is compared to other segments. This comparison is done regardless of the selected coding mode and .

 ()

Thus, if the energy in any of the above defined segments, other than , is close to that of the candidate attack, the attack is eliminated by setting  to 0.

Initially the speech/music decision in the second stage is set equal to the speech/music decision from the first stage, i.e. . In case the decision is “music”, it could be reverted to “speech” in the following situations.

The decision is reverted from music to speech for highly correlated stable signals with higher pitch period. These signals are characterized by

 ()

Further, if the above condition is fulfilled and the selected coding mode was TC, it is changed to GC. This is to avoid any transition artefacts during stable harmonic signal.

In case there is an energetic event characterized by  and at the same time  it could mean that an attack has occurred in the input signal and the following logic takes place. If, in this situation, the counter of frames from the last detected onset/transition , described in subclause 5.1.13.4, has been set to 1 the attack is confirmed and the decision is changed to speech, i.e. . Also, the coding mode is changed to TC. Otherwise, if there has been an attack found by the attack tracking algorithm described above, and the position of this attack is beyond the end of the third sub-frame, the decision is also changed to speech and the coding mode is changed to TC. That is

 ()

Furthermore, an attack flag  is set to 1 if the detected attack is located after the first quarter of the first sub-frame, i.e. when . This flag is later used by the GSC technology. Finally, the attack flag  is set to 1 in all active frames () that have been selected for GC coding and for which the decision in the first stage of the speech/music classifier was “speech”. However, it is restricted only to frames in which the attack is located in the fourth subframe. In this case, the coding mode is also changed to TC for better representation of the attack.

As previously described, if =flag\_spitch=1, VC mode is maintained and AC mode is set to 0; that is,

if (=1 and sampling rate = 16kHz and bit rate < 13.2kbps )

{

=0;

}

##### 5.1.13.6.7 Context-based improvement of the classification for stable tonal signals

By using context-based improvement of the classification, an error in the classification in the previous stage can be corrected. If the current frame has been provisionally classified as “speech”, the classification result can be corrected to “music”, and vice versa. To determine a possible error in the current frame, the values of 8 consecutive frames including the current frame are considered for some features.

Figure shows the multiple coding mode signal classification method. If the current frame has been provisionally classified as “speech” after the first- and second-stage classification, then the frame is encoded using the CELP-based coding. On the other hand, if the current frame is initially classified as “music” after the first- and second-stage classification, then the frame is further analysed for fine-classification of “speech” or “music” to select either the GSC-based coding or MDCT-based transform coding, respectively. The parameters used to perform the fine-classification in multiple coding mode selection include:

* Tonality
* Voicing
* Modified correlation
* Pitch gain, and
* Pitch difference

The tonality in the sub-bands of 0-1kHz, 1-2 kHz, and 2-4 kHz are estimated as , , and  as follows:







where  is the power spectrum. The maximum tonality, , is estimated as,



The voicing feature, , is the same one as used in the unvoiced signal classification. See equation (237) in subclause 5.1.13.1.1. for the details of its computation. The voicing feature from the first analysis window is used, i.e.



The modified correlation, , is the normalized correlation from the previous frame.

The pitch gain, , is the smoothed closed-loop pitch gain estimated from the previous frame, i.e.,



where  is the ACB gain in each of the sub-frames from the previous frame.

The pitch deviation is estimated as the sum of pitch differences between the current frame open-loop pitch ,  and the open loop pitch in the previous three frames, 



The features, , , and  are smoothed to minimize spurious instantaneous variations as follows: 





where  and are 0.1 in active frames (i.e., SAD = 1), and 0.7 in background noise and inactive frames. Similarly, is 0.1 in active frames and 0.5 in inactive frames.



Figure 18 : Multiple coding mode signal classification

The following condition is evaluated to select the GSC or MDCT based coding,



A hangover logic is used to prevent frequent switching between coding modes of GSC and MDCT-based coding. A hangover period of 6 frames is used. The coding mode is further modified as per below.

Figure 19 shows two independent state machines, which are defined in the context-based classifier, SPEECH\_STATE and MUSIC\_STATE. Each state machine has two states . In each state a hangover of 6 frames is used to prevent frequent transitions. If there is a change of decision within a given state, the hangover in each state is set to 6, and the hangover is then reduced by 1 for each subsequent frame. A state change can only occur once the hangover has been reduced to zero. The following six features are used in the context-based classifier (the superscript [-*i*] is used below to denote the past frames).

The tonality in the region of 1~2 kHz,  is defined as

 ()

The tonality in the region of 2~4 kHz,  is defined as

 ()

The long-term tonality in the low band,  is defined as

 ()

The difference between the tonality in 1~2 kHz band and the tonality in 2~4 kHz band is defined as

 ()

The linear prediction error  is defined as

 ()

where  has been defined in equation (327).

The difference between the scaled voicing feature  defined equation (327) and the scaled correlation map feature  defined in equation (327) is defined as

 ()

The following two independent state machines are used to correct errors in the previous stages of the speech/music classification. The are two state machines are called SPEECH\_STATE and MUSIC\_STATE. There are also two hangover variables denoted  and  which are initialized to the value of 6 frames. The following four conditions are evaluated to determine the transition of one state to another.

Condition A is defined as

 ()

Condition B is then defined as

 ()

Condition C is defined as

 ()

and finally condition D is defined as

 ()



Figure 19 : State machines for context-based speech/music correction

The decisions from the speech/music classifier,  and  are changed to 0 (“speech”) if  was previously set to 1 (“music”) and if the context-based classifier is in SPEECH\_STATE. Similarly, the decisions from the speech/music classifier,  and  are changed to 1 (“music”) if  was previously set to 0 (“speech”) and if the context-based classifier is in MUSIC\_STATE.

##### 5.1.13.6.8 Detection of sparse spectral content

At 13.2kbps, the coding of music signal benefits from combining the advantages of MDCT and GSC technologies. For frames classified as music after the context-based improvement, coding mode producing better quality is selected between MDCT and GSC based on an analysis of signal spectral sparseness and linear prediction efficiency (depending on the input bandwidth).

For each active frame, the sum of the log energy spectrum  is calculated to determine the spectral sparseness analysis.

 ()

Then the log energy spectrum  is sorted in descending order of magnitude. Each element in the sorted log energy spectrum  is accumulated one by one along in descending order until the accumulated value exceeds 75% of the . The index (or the position),, of the last element added into the accumulation can be regarded as a kind of representation of the spectral sparseness of the frame and is stored into a sparseness buffer  of 8 frames.

If the input bandwidth is WB, some parameters dedicated to WB are calculated, including the mean of the sparseness buffer, the long-term smoothed sparseness, the high-band log energy sum, the high-band high sparseness flag, the low-band high sparseness flag, the linear prediction efficiency and the voicing metric. For other input bandwidths the above parameters are not calculated. The mean of the sparseness buffer is obtained as

 ()

Then the long-term smoothed sparseness  is calculated as

 ()

where denotes the long-term smoothed sparseness in the previous frame,  denotes the average of the four smallest values in the sparseness buffer . The reason of using  is to reduce the possible negative impact to the  from interfering frames. The long-term smoothed sparseness  is initialized to the value of  and the sparseness buffer  is also initialized to the value of  for all its elements if the current frame is the first active frame after a pause. The high-band log energy sum is calculated over 

 ()

To obtain the high-band high sparseness flag, the high-band log energy spectrum  is first sorted in descending order. The ratio of the sum of the first 5 elements (or the 5 largest values) of the sorted high-band log energy spectrum to the high-band log energy sum is calculated

 ()

where  is the sorted high-band log energy spectrum. The ratio  can be regarded as a kind of representation of the high-band spectral sparseness of the frame and is stored into a high-band sparseness buffer . The mean of the buffer  is calculated. If the mean is greater than 0.2, the high-band high sparseness flag  is set to 1 indicating a high sparseness of the high-band spectrum, otherwise set to 0. Similarly, to obtain the low-band high sparseness flag, the low-band log energy spectrum  is sorted in descending order and the ratio of the sum of the first 5 elements (or the 5 largest values) of the sorted low-band log energy spectrum to the low-band log energy sum is calculated

 ()

where  is the sorted low-band log energy spectrum. The ratio  can be regarded as a kind of representation of the low-band sparseness of the frame. If the ratio is greater than 0.18, the low-band high sparseness flag  is set to 1 indicating a high sparseness of the low-band spectrum, otherwise set to 0. The LP residual log-energy ratio of the current frame , as calculated in subclause 5.1.13.5.1 which is shown again below

 ()

is stored into a LP residual log-energy ratio buffer  of 8 frames. The mean of the buffer , , is calculated and used to represent the short-term linear prediction efficiency at the current frame. The lower the  is the higher the short-term linear prediction efficiency is. The scaled normalized correlation as calculated in subclause 5.1.13.5.2 is stored into a voicing buffer  of 8 frames. The mean of the buffer , , is calculated and used to represent the voicing metric at the current frame.

Decision on which coding mode to use (MDCT or GSC) is made for each frame previously classified as music, that is, for each frame where  is set to 1. GSC coding mode is selected by setting  to 1 and changing  to 0. GSC coding mode is selected for frame with extremely non-sparse spectrum, that is, when  is greater than 90. In this case, the GSC hangover flag  is also set to 1 meaning that a soft hangover period will be applied. Otherwise, if  is set to 1, the current frame is in a soft hangover period where the determination of extremely non-sparse spectrum is slightly relaxed, that is, GSC coding mode is selected if  is greater than 85. If in above case,  is not greater than 85, GSC coding mode is still selected if  of the current frame is deviating from the average  of its adjacent GSC coded frames by less than 7. A maximum of 7 frames are used for the averaging. The selection between MDCT coding mode and GSC coding mode ends here if the input bandwidth is SWB. For WB input bandwidth, one more step is applied. In this case, GSC coding mode is also selected if the various sparseness measures calculated all do not exhibit strong sparseness characteristics and the linear prediction efficiency is assumed high. Specifically, GSC coding mode is selected if  and  and  and  and  is set to 0 and  or if condition  is not met but is not set to 1. In above case, the GSC hangover flag  is also set to 1. The flag  is set to 0 if GSC coding mode is not selected through the whole procedure described above.

##### 5.1.13.6.9 Decision about AC mode

The decisions in the first and in the second stage of the speech/music classifier, refined and corrected by the modules described so far are used to determine the usage of the AC mode. As mentioned before, in the AC mode GSC technology is used to encode the input signal. The decision about the AC mode is done always but the GSC technology is used only at certain bitrates. This is described in subclause 5.1.16.

Before making decision about the AC mode, the speech/music classification results are overridden for certain noisy speech signals. If the level of the background noise is higher than 12dB, i.e. when , then . This is a protection against mis-classification of active noisy speech signals.

For certain unvoiced SWB signals, GSC technology is preferred over the UC or GC mode which would normally be selected. In order to override the selection of the coding mode, there is a flag denoted  which is set to 1 under the following condition

 ()

where  is the raw coding mode, described in subclause 5.1.13.2.

The AC mode is selected if  or if .

##### 5.1.13.6.10 Decision about IC mode

The IC mode has been designed and optimized for inactive signals which are basically the background noise. Two encoding technologies are used for the encoding of these frames, the GSC and the AVQ. The GSC technology is used at bitrates below 32kbps and the AVQ is used otherwise. The selection of the IC mode at bitrates below 32kbps are conditioned by  whereas for higher bitrates, the condition is changed to .

The TC mode and the AC mode are not used at 9.6, 16.4 and 24.4 kbps. Thus, at these bitrates, the coding mode is changed to the GC mode if it was previously set to the TC or AC mode. Furthermore, the selection of the IC mode at the previously mentioned bitrates is conditioned only by .

### 5.1.14 Coder technology selection

Multiple coding technologies are employed within the EVS codec, based on one of the following two generic principles for speech and audio coding, the LP-based (analysis-by-synthesis) approach and the transform-domain (MDCT) approach. There is no clearly defined borderline between the two approaches in the context of this codec. The LP-based coder is essentially based on the CELP technology, optimized and tuned specifically for each bitrate. The transform-domain approach is adopted by the HQ MDCT technology. There are also two hybrid schemes in which both approaches are combined, the GSC technology and the TCX technology. The selection of the coder technology depends on the actual bitrate, the bandwidth, speech/music classification, the selected coding mode and other parameters. The following table shows the allocation of technologies based on bitrate, bandwidth and content.

Table 19: Allocation of coder technologies per bitrate, bandwidth and content

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| bitrate | 7.2 | 8 | 9.6 | 13.2 | 16.4 | 24.4 | 32 | 48 | 64 |
| NB |  |  |  |  |  |  |  |  |  |
| speech | ACELP | ACELP | ACELP | ACELP | ACELP | ACELP |  |  |  |
| audio | HQ MDCT | HQ MDCT | TCX | TCX/HQ MDCT | TCX/HQ MDCT | TCX |  |  |  |
| noise | GSC | GSC | TCX | GSC | TCX | TCX |  |  |  |
| WB |  |  |  |  |  |  |  |  |  |
| speech | ACELP | ACELP | ACELP | ACELP | ACELP | ACELP | ACELP | TCX | ACELP |
| audio | GSC | GSC | TCX | GSC/TCX/HQ MDCT | TCX/HQ MDCT | TCX | HQ MDCT | TCX | HQ MDCT |
| noise | GSC | GSC | TCX | GSC | TCX | TCX | ACELP | TCX | ACELP |
| SWB |  |  |  |  |  |  |  |  |  |
| speech |  |  |  | ACELP | ACELP | ACELP | ACELP | TCX | ACELP |
| audio |  |  |  | GSC/TCX/HQ MDCT | TCX/HQ MDCT | TCX/HQ MDCT | TCX/HQ MDCT | TCX | HQ MDCT |
| noise |  |  |  | GSC | TCX | TCX | ACELP | TCX | ACELP |
| FB |  |  |  |  |  |  |  |  |  |
| speech |  |  |  |  | ACELP | ACELP | ACELP | TCX | ACELP |
| audio |  |  |  |  | TCX | TCX/HQ MDCT | TCX/HQ MDCT | TCX | HQ MDCT |
| noise |  |  |  |  | TCX | TCX | ACELP | TCX | ACELP |

The TCX technology is used for any content at bitrates higher than 64 kbps.

At 9.6kbps, 16.4kbps and 24.4kbps a specific technology selector is used to select either ACELP or an MDCT-based technology (HQ MDCT or TCX). This selector is described in clause 5.1.14.1.

At all other bitrates, the division into “speech”, “audio” and background “noise” is based on the decision of the SAD and on the decision of the speech/music classifier.

The decision between the TCX technology and the HQ MDCT technology is done adaptively on a frame-by-frame basis. There are two selectors, one for 13.2 and 16.4 kbps and the second for 24.4 and 32 kbps. There is no adaptive selection beyond these bitrates as shown in the above table. These two selectors are described in detail in the subclauses 5.1.14.2 and 5.1.14.3.

#### 5.1.14.1 ACELP/MDCT-based technology selection at 9.6kbps, 16.4 and 24.4 kbps

At 9.6kbps, 16.4kbps and 24.4kbps the decision to choose either ACELP or an MDCT-based technology is not based on the decision of the speech/music classifier as it is done for other bitrates, but on a specific technology selector described below.

The technology selector is based on two estimates of the segmental SNR, one estimate corresponding to the transform-based technology (described in subclause 5.1.14.1.1), another estimate corresponding to the ACELP technology (described in subclause 5.1.14.1.2). Based on these two estimates and on a hysteresis mechanism, a decision is taken (described in subclause 5.1.14.1.3).

##### 5.1.14.1.1 Segmental SNR estimation of the MDCT-based technology

The segmental SNR estimation of the TCX technology is based on a simplified TCX encoder. The input audio signal is first filtered using a LTP filter, then windowed and transformed using a MDCT, the MDCT spectrum is then shaped using weighted LPC, a global gain is then estimated, and finally the segmental SNR is derived from the global gain. All these steps are described in detail in the following clauses.

5.1.14.1.1.1 Long term prediction (LTP) filtering

The LTP filter parameters (pitch lag and gain) are first estimated. The LTP parameters are not only used for filtering the audio input signal for estimating the segmental SNR of the transform-based technology. The LTP parameters are also encoded into the bitstream in case the TCX coding mode is selected, such that the TCX LTP postfilter described in subclause 6.9.2.2 can use them. Note that the LTP filter parameter estimation is also performed at 48kbps, 96kbps and 128kbps even though the parameters are not used to filter the audio input signal in this case.

A pitch lag with fractional sample resolution is determined, using the open-loop pitch lag  and an interpolated autocorrelation. The LTP pitch lag has a minimum value of , a maximum value of  and a fractional pitch resolution . Additionally, two thresholds  and  are used. If the pitch lag is less than , the full fractional precision  is used. If the pitch lag is greater than , no fractional lag is used. For pitch lags in between, half of the fractional precision  is used. These parameters depend on the bitrate and are given in the table below.

Table 20: LTP parameters vs bitrate

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Bitrate | Bandwidth | LTP sampling rate | LTP frame length |  |  |  |  |  |
| 9.6 kbps | NB, WB, SWB | 12.8kHz | 256 | 4 | 29 | 231 | 154 | 121 |
| 16.4-24.4 kbps | NB | 12.8kHz | 256 | 4 | 29 | 231 | 154 | 121 |
| 16.4-24.4 kbps | WB, SWB, FB | 16kHz | 320 | 6 | 36 | 289 | 165 | 36 |
| 48 kbps | WB, SWB, FB | 25.6kHz | 512 | 4 | 58 | 463 | 164 | 58 |
| 96-128 kbps | WB, SWB, FB | 32kHz | 640 | 6 | 72 | 577 | 75 | 72 |

First the parameter  is initialized depending on the fractional pitch resolution:

. ()

Then the search range for the pitch lag is determined as follows:

. ()

For the search range, autocorrelation of the weighted input signal  (including the look-ahead part) is computed (note that at 48kbps, 96kbps and 128kbps, the weighted input signal is not available, so the non-weighted input signal is used instead), extended by 4 additional samples in both directions required for subsequent interpolation filtering:

. ()

Within the search range, the index and value of the maximum correlation are determined:

. ()

The maximum correlation value is normalized as follows:

. ()

The fractional precision of the transmitted pitch lag is determined by the initial pitch lag , the maximum fractional resolution , and the thresholds  and .

. ()

For determining fractional pitch lag the autocorrelation  is interpolated around the maximum value by FIR filtering:

. ()

. ()

. ()

The integer and fractional parts of the refined pitch lag ( and ) are then determined by searching the maximum of the interpolated correlation:

. ()

For transmission in the bitstream, the pitch lag is encoded to an integer index  (that can be encoded with 9 bits) as follows:

. ()

The decision if LTP is activated is taken according to the following condition:

. ()

with the temporal flatness measure  and the maximum energy change  are computed as described in clause 5.1.8.

If LTP is activated, the predicted signal  is computed from the input signal  (including the lookahead part) by interpolating the past input signal using a polyphase FIR filter. The polyphase index of the filter is determined by the fractional pitch lag:

. ()

. ()

. ()

The LTP gain  is computed from the input and predicted signals:

. ()

For transmission in the bitstream, the gain is quantized to an integer index  (that can be encoded with 2 bits)

. ()

The quantized gain  is computed as:

. ()

If the quantized gain is less than zero, LTP is deactivated:

. ()

If LTP is not active, the LTP parameters are set as follows:

. ()

The LTP filtered signal is then computed, except at 48kbps, 96kbps and 128kbps. The LTP filtered signal is computed by multiplying the predicted signal with the LTP gain and subtracting it from the input signal. To smooth parameter changes, a zero input response is added for a 5ms transition period. If LTP was not active in the previous frame, a linear fade-in is applied to the gain over a 5ms transition period.

If LTP was active in the previous frame, the zero input response  is computed:

. ()

. ()

The zero input response is then computed by LP synthesis filtering with zero input, and applying a linear fade-out to the second half of the transition region:

. ()

. ()

. ()

with the LP coefficients are obtained by converting the mid-frame LSP vector of the current frame  using the algorithm described in subclause 5.1.9.7. Finally the LTP filtered signal is computed:

. ()

5.1.14.1.1.2 Windowing and MDCT

The LTP filtered signal  is windowed using a sine-based window whose shape depends on the previous mode. If the past frame was encoded with a MDCT-based coding mode, the window is defined as

, for . ()

, for . ()

, for . ()

, for . ()

, for . ()

If the past frame was encoded with the ACELP coding mode, the window is defined as

, for . ()

, for . ()

, for . ()

, for . ()

with ,  at 12.8kHz, and ,  at 16kHz. The total length of the window is  (40ms) when the past frame was encoded with a MDCT-based coding mode and  (50ms) when the past frame was encoded with the ACELP coding mode.

The windowed LTP-filtered signal is transformed with a MDCT using time domain aliasing (TDA) and a discrete cosine transform (DCT) IV as described in subclause 5.3.2.2, producing the MDCT coefficients with ,  is  when the past frame was encoded with a MDCT-based coding mode and  is  when the past frame was encoded with the ACELP coding mode.

5.1.14.1.1.3 MDCT spectrum shaping

The mid-frame LSP vector of the current frame  is converted into LP filter coefficients  using the algorithm described in clause 5.1.9.7. The LP filter coefficients  are then weighted as described in clause 5.1.10.1, producing weighted LP filter coefficients with at 12.8kHz and at 16kHz. The weighted LP filter coefficients are then transformed into the frequency domain as described in subclause 5.3.3.2.3.2. The obtained LPC gains are finally applied to the MDCT coefficients as described in subclause 5.3.3.2.3.3, producing the LPC shaped MDCT coefficients .

When the encoded bandwidth is NB, the MDCT coefficients corresponding to the frequencies above 4kHz are set to zeros:, .

5.1.14.1.1.4 Global gain estimation

A global gain  is estimated similarly to the first step described in subclause 5.3.3.2.8.1.1. The energy of each block of 4 coefficients is first computed:

. ()

A bisection search is performed with a final resolution of 0.125dB:

**Initialization:** Set *fac* = *offset = 128* and *target = 500 if NB, target = 850 otherwise*

**Iteration:** Do the following block of operations 10times

1- *fac=fac/2*

2- *offset = offset – fac*

2- 

3- *if(ener>target) then offset=offset+fac*

If *offset<=32, then offset=* -128.  
The gain is then given by:



5.1.14.1.1.5 Segmental SNR estimation of the MDCT-based technology

The estimated TCX SNR in one subframe is given by

. ()

Finally, the estimated segmental SNR of the whole encoded TCX frame is obtained by converting the per-subframe SNRs into dB and averaging them over all subframes.

##### 5.1.14.1.2 Segmental SNR estimation of the ACELP technology

The segmental SNR estimation of the ACELP technology is based on the estimated SNR of the adaptive-codebook and the estimated SNR of the innovative-codebook. This is described in detail in the following clauses.

5.1.14.1.2.1 SNR estimation of the adaptive-codebook

An integer pitch-lag per subframe  is derived from the refined open-loop pitch lags  (see clause 5.1.10.9).  
When the sampling-rate is 12.8kHz, the number of subframe is four, and the integer pitch lags are simply equal to the refined open-loop pitch lags rounded to the nearest integer.  
When the sampling-rate is 16kHz, the number of subframe is five. The refined open-loop pitch lags are first scaled by a factor of 1.25, then they are rounded to the nearest integer and finally the four obtained integer pitch lags are mapped to the five subframes. The first integer pitch-lag is mapped to the first subframe, the second integer pitch-lag is mapped to the second subframe, the third integer pitch-lag is mapped to the third and fourth subframes, and the fourth integer pitch-lag is mapped to the fifth subframe.

A gain is then computed for each subframe

. ()

The estimated SNR of the adaptive-codebook is then computed for each subframe

. ()

5.1.14.1.2.2 SNR estimation of the innovative-codebook

The estimated SNR of the innovative-codebook  is assumed to be a constant, which depends on the encoded bandwidth and on the bitrate.  at 9.6kbps NB,  at 9.6kbps WB and  at 16.4 and 24.4kbps WB and SWB.

5.1.14.1.2.3 Segmental SNR estimation of ACELP

The estimated SNR of one ACELP encoded subframe is then computed by combining the adaptive-codebook SNR and the innovative-codebook SNR.

. ()

Finally, the estimated segmental SNR of the whole encoded ACELP frame is obtained by converting the per-subframe SNRs into dB and averaging them over all subframes.

##### 5.1.14.1.3 Hysteresis and final decision

The ACELP technology is selected if

. ()

otherwise the MDCT-based technology is selected.

adds hysteresis in the decision, in order to avoid switching back and forth too often between the two coding technologies. is computed as described below ( is 0 by default). Further, in 12.8 kHz core (i.e., 9.6 kbps and 13.2 kbps), the dssnr is updated as shown in equation (433a).

. (432)

. (433)

. (433a)

where  is described in clause 5.1.14.1.1.4, and are described in clause 5.1.13.6, and  is described in 5.1.11.2.1.

. (434)

with  is the temporal flatness measure described in clause 5.1.8,  is a stability factor described in subclause 6.1.1.3.2 but using the unquantized LSF parameters estimated at 12.8kHz,is the number of consecutive previous ACELP frames (if the previous frame was not ACELP, ),  is the long-term SNR as described in clause 5.1.12,  is the SAD decision as described in clause 5.1.12, and  indicates whether DTX is enabled or not.

#### 5.1.14.2 TCX/HQ MDCT technology selection at 13.2 and 16.4 kbps

The selection between TCX and HQ MDCT (Low Rate HQ) technology at 13.2 kbps (NB, WB and SWB) and 16.4 kbps (WB and SWB) is done on a frame-by-frame basis and is based on the following measures.

– Voicing measures

– Spectral noise floor

– SAD decision

– High-band energy

– High-band sparseness (with hysteresis)

The boundaries of frequency bands for the purposes of the TCX/HQ technology selection is set according to the following table.

Table 21: Boundaries of frequency bands for TCX/HQ MDCT (Low Rate HQ) selection

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Band width | Low band CLDFB | High band CLDFB | Low band FFT | High band FFT |
| NB | 8 | 10 | /4 | \*5/16 |
| WB | 12 | 20 | \*3/8 | /2 |
| SWB | 16 | 40 | /2 for sparseness,  \*3/8 otherwise | /2 |

Voicing measure is defined as the average of pitch gain  of the former half-frame and of the latter half-frame defined in (81),

. ()

Sparseness measure  is defined as

, ()

whereis a number of bins which attain following condition within low band:

, ()

where is an averaged energy of all spectrum bands.

High energy measureis defined in terms of CLDFB energy as

. ()

Flag indicating the sparseness for high bands, = TRUE when

, ()

where is a number of FFT bins within and  which attain

. ()

Otherwise, =FALSE.

Flag indicating the sparseness for high bands with hysteresis, = TRUE when

. ()

Otherwise, =FALSE.

Additionally,  is set TRUE when following is satisfied:

. ()

is the averaged energy only for the local minima of the spectrum. With the notation of 5.1.11.2.5, it is defined as:

. ()

Correlation map sum,  is defined in 5.1.11.2.5.

Indication of possible switching, =TRUE when previous core was not Transform coding, or followings are satisfied.

, (444)

where and  are  and  at the previous frames. Note that is integer from -1 to 2, while others are all Boolean.

Indication of preference for TCX,  = TRUE when followings are satisfied:

 (445)

Indication of preference for HQ MDCT,  = TRUE when followings are satisfied:

, ()

where *transient\_frame* is the output of the time-domain transient detector (see 5.1.8). For 16.4 kbps,  is set to FALSE and  to TRUE when *transient\_frame* is detected.

Based on the above definitions and thresholds listed in the table below, switching between HQ and MDCT based TCX is carried out as follows. Switching between HQ and TCX can only occur when  is TRUE. In this case, TCX is used if  is TRUE, or otherwise HQ is used if  is TRUE. In any other case, the same kind of transform coding is applied as in the previous frame. If the previous frame was not coded by transform coding, HQ is used for the low rate (13.2 kbps) and TCX for the high rate (16.4 kbps).

In case input signal is noisy speech (*noisy\_speech\_flag*==TRUE && *vadflag*== FALSE) , transition from TCX to HQ is prohibited at 16.4 kbps.

 is reset to 0 if  is FALSE, otherwise it is incremented by one (with a maximum allowed value of 2)

 and  are reset to FALSE and -1, respectively, upon encoder initialization or when a non-transform-coded frame is encountered.

Table : List of thresholds used in TCX/HQ MDCT (Low Rate HQ) selection

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Meaning | 13.2 kbps | 16.4 kbps |
| SIG\_LO\_LEVEL\_THR | Low level signal | 22.5 | 23.5 |
| SIG\_HI\_LEVEL\_THR | High level signal | 28.0 | 19.0 |
| COR\_THR | correlation | 80.0 | 62.5 |
| VOICING\_THR | voicing | 0.6 | 0.4 |
| SPARSENESS\_THR | sparseness | 0.65 | 0.4 |
| HI\_ENER\_LO\_THR | High energy low limit | 9.5 | 12.5 |
| HYST\_FAC | Hysteresis control | 0.8 | 0.8 |
| MDCT\_SW\_SIG\_LINE\_THR | Significant Spectrum | 2.85 | 2.85 |
| MDCT\_SW\_SIG\_PEAK\_THR | Significant peak | 36.0 | 36.0 |

#### 5.1.14.3 TCX/HQ MDCT technology selection at 24.4 and 32 kbps

The decision between using the TCX technology or the HQ MDCT (high rate HQ) technology at 24.4 kbps and 32 kbps for SWB signals is based on the average energy values and peak-to-average ratios of different sub-bands, furthermore, the average energy values and peak-to-average ratios are calculated by the CLDFB band energy analysis , spectral analysisand the bit-rate.

First, the average energy of three CLDFB sub-bands: 0~3.2kHz, 3.2~6.4kHz and 6.4~9.6kHz ,  are calculated according to

 ()

Second, the spectral peak  and spectral average ,  of the FFT sub-bands: 1~2.6kHz and 4.8~6.4kHz are calculated according to

 ()

At 24.4kbps, the CLDFB sub-band (4.8~9.6kHz) average energy , and the CLDFB sub-band (400-3.2kHz) average energy  are also calculated according to

 ()

The peak energy  and average energy  of the CLDFB sub-band (8~10kHz) are also calculated according to

 ()

To identify the MDCT coding mode, three conditions are identified:

Condition I:

 ()

Condition II:

 ()

Condition III:

 ()

The primary classifier decision  at 24.4kbps is formed according to

 ()

At 32kbps, further spectral analysis is needed. First, a noise-floor envelope and a peak envelope are calculated as

 (455)

and

 (456)

respectively, where the smoothing factors  and depend on the instantaneous magnitude spectrum

 ()

 ()

The noise-floor energy and the peak envelope energy are formed by averaging the noise-floor and peak envelopes, respectively. That is,

 ()

 ()

Spectral peaks are identified in two steps. First, all for which holds true are marked as peak candidates. Second, for each sequence of consecutive, the largest spectral magnitude is kept as a peak representative for that sequence. Peak sparseness measure is formed by averaging the peak distances among the peak representatives, with if less than 2 peaks are identified. Two decision variables are formed

 ()

The peak energy  and average energy  of the CLDFB sub-band at10~12 kHz are calculated according to

 ()

Three conditions are then checked.

Condition I:

 ()

Condition II:

 ()

Condition III:

 (465)

The primary classifier decision  at 32kbps is formed according to

 (466)

To increase the classifier stability for both 24.4kbps and 32kbps, the primary classifier decision is low-pass filtered from frame to frame.

 (467)

Finally, hysteresis is applied such that the classifier decision from the previous frame is only changed if the decision passes the switching range 

 (468)

If none of these conditions are met, the previous classifier is kept, i.e. .and the buffers are updated as follows

 (469)

#### 5.1.14.4 TD/Multi-mode FD BWE technology selection at 13.2 kbps and 32 kbps

The input WB or SWB signal is divided into low band signal and high band signal (wideband input) or super higher band signal (super wideband input). Firstly, the low band signal is classified based on the characteristics of the low band signal and coded by the LP-based approach or the transform-domain approach.

The selection between TD BWE and multi-mode FD BWE technology of super higher band signal or high band signal at 13.2 kbps (WB and SWB) and 32 kbps (SWB) is performed based on the characteristic of the input signal and coding modes of the low band signal. Except for MDCT mode, if the input signal is classified as music signal, the high band signal or the super higher band signal is encoded by multi-mode FD BWE;if the input signal is classified as speech signal, the high band signal or the super higher band signal is encoded by TD BWE. In the case that the low band signal is classified as IC mode, the high band signal or the super higher band signal is also encoded by multi-mode FD BWE.

If the decision in the first stage of the speech/music classifier, i.e. the input signal is classified as music signal, or the decision in the first stage of the speech/music classifierand the decision in the second stage of the speech/music classifier, or the low band signal is classified as IC mode, the high band or the super higher band signal is encoded by multi-mode FD BWE, otherwise, the high band or super higher band signal is encoded by TD BWE. It is noted that, when the flag of the super wideband noisy speech, the super higher band is encoded by TD BWE. It is the same TD/multi-mode FD BWE technology selection for FB inputs.