EUROPEAN TELECOMMUNICATION STANDARD

ETS 300 581-6

November 1995

Source: ETSI TC-GSM

Reference: DE/SMG-020642

ICS: 33.060.50

Key words: European digital cellular telecommunications system, Global System for Mobile communications (GSM)

European digital cellular telecommunications system; Half rate speech Part 6: Voice Activity Detector (VAD) for half rate speech traffic channels (GSM 06.42)

ETSI

European Telecommunications Standards Institute

ETSI Secretariat

Postal address: F-06921 Sophia Antipolis CEDEX - FRANCE
Office address: 650 Route des Lucioles - Sophia Antipolis - Valbonne - FRANCE
X.400: c=fr, a=atlas, p=etsi, s=secretariat - **Internet:** secretariat@etsi.fr

Tel.: +33 92 94 42 00 - Fax: +33 93 65 47 16

*

Copyright Notification: No part may be reproduced except as authorized by written permission. The copyright and the foregoing restriction extend to reproduction in all media.

© European Telecommunications Standards Institute 1995. All rights reserved.

Whilst every care has been taken in the preparation and publication of this document, errors in content, typographical or otherwise, may occur. If you have comments concerning its accuracy, please write to "ETSI Editing and Committee Support Dept." at the address shown on the title page.

Contents

Blank page

Foreword

This European Telecommunication Standard (ETS) has been produced by the Special Mobile Group (SMG) Technical Committee of the European Telecommunications Standards Institute (ETSI).

This ETS specifies the half rate speech traffic channels for the European digital cellular telecommunications system. This ETS corresponds to GSM technical specification, GSM 06.42, version 4.1.1 and is part 6 of a multi-part ETS covering the half rate speech traffic channels as described below:

GSM 06.02	ETS 300 581-1: "European digital cellular telecommunications system; Half rate speech Part 1: Half rate speech processing functions".
GSM 06.20	ETS 300 581-2: "European digital cellular telecommunications system; Half rate speech Part 2: Half rate speech transcoding".
GSM 06.21	ETS 300 581-3: "European digital cellular telecommunications system; Half rate speech Part 3: Substitution and muting of lost frames for half rate speech traffic channels".
GSM 06.22	ETS 300 581-4: "European digital cellular telecommunications system; Half rate speech Part 4: Comfort noise aspects for half rate speech traffic channels".
GSM 06.41	ETS 300 581-5: "European digital cellular telecommunications system; Half rate speech Part 5: Discontinuous Transmission (DTX) for half rate speech traffic channels".
GSM 06.42	**ETS 300 581-6: "European digital cellular telecommunications system (Phase 2); Half rate speech Part 6: Voice Activity Detection (VAD) for half rate speech traffic channels".**
GSM 06.06	ETS 300 581-7: "European digital cellular telecommunications system; Half rate speech Part 7: ANSI-C code for the GSM half rate speech codec".
GSM 06.07	ETS 300 581-8: "European digital cellular telecommunications system; Half rate speech Part 8: Test vectors for the GSM half rate speech codec".

NOTE: TC-SMG has produced documents which give the technical specifications for the implementation of the European digital cellular telecommunications system. Historically, these documents have been identified as GSM Technical Specifications (GSM-TS). These TSs may have subsequently become Interim European Telecommunication Standards (I-ETSs), (Phase 1), or European Telecommunication Standards (ETSs), (Phase 2), whilst others may become ETSI Technical Reports (ETRs).

Transposition dates	
Date of adoption of this ETS:	27 October 1995
Date of latest announcement of this ETS (doa):	28 February 1996
Date of latest publication of new National Standard or endorsement of this ETS (dop/e):	31 August 1996
Date of withdrawal of any conflicting National Standard (dow):	31 August 1996

Blank page

1 Scope

This European Telecommunication Standard (ETS) specifies the Voice Activity Detector (VAD) to be used in the Discontinuous Transmission (DTX) as described in GSM 06.41 (ETS 300 581-5) [4]. It also specifies the test methods to be used to verify that a VAD implementation complies with this ETS.

The requirements are mandatory on any VAD to be used either in GSM Mobile Stations (MS)s or Base Station Systems (BSS)s that utilise the half-rate GSM speech traffic channel.

2 Normative references

This ETS incorporates by dated and undated reference, provisions from other publications. These normative references are cited at the appropriate places in the text and the publications are listed hereafter. For dated references, subsequent amendments to or revisions of any of these publications apply to this ETS only when incorporated in it by amendment or revision. For undated references, the latest edition of the publication referred to applies.

[1] GSM 01.04 (ETR 100): "European digital cellular telecommunications system; Abbreviations and acronyms".

[2] GSM 06.20 (ETS 300 581-2): "European digital cellular telecommunications system; Half rate speech Part 2: Half rate speech transcoding".

[3] GSM 06.22 (ETS 300 581-4): "European digital cellular telecommunications system; Half rate speech Part 4: Comfort noise aspects for half rate speech traffic channels".

[4] GSM 06.41 (ETS 300 581-5): "European digital cellular telecommunications system; Half rate speech Part 5: Discontinuous transmission (DTX) for half rate speech traffic channels".

[5] GSM 06.06 (ETS 300 581-7): "European digital cellular telecommunications system; Half rate speech Part 7: ANSI C code for the GSM half rate speech codec".

[6] GSM 06.07 (ETS 300 581-8): "European digital cellular telecommunications system; Half rate speech Part 8: Test sequences for the GSM half rate speech codec".

3 Definitions, symbols and abbreviations

3.1 Definitions

For the purpose of this ETS, the following definitions apply.

noise: The signal component resulting from acoustic environmental noise.

mobile environment: Any environment in which mobile stations may be used.

3.2 Symbols

For the purpose of this ETS, the following symbols apply.

3.2.1 Variables

aav1	filter predictor values, see subclause 5.2.3
acf	the ACF vector which is calculated in the speech encoder (GSM 06.20 (ETS 300 581-2) [2])
adaptcount	secondary hangover counter, see subclause 5.2.6
av0	averaged ACF vector, see subclause 5.2.2
av1	a previous value of av0, see subclause 5.2.2

burstcount	speech burst length counter, see subclause 5.2.7
den	denominator of left hand side of equation 8 in annex C, see subclause 5.2.5
difference	difference between consecutive values of dm, see subclause 5.2.4
dm	spectral distortion measure, see subclause 5.2.4
hangcount	primary hangover counter, see subclause 5.2.7
lagcount	number of subframes in current frame meeting periodicity criterion, see subclause 5.2.9
lastdm	previous value of dm, see subclause 5.2.4
lags	the open loop long term predictor lags for the four speech encoder subframes (GSM 06.20 (ETS 300 581-2) [2].)
num	numerator of left hand side of equation 8 in annex C, see subclause 5.2.5
oldlagcount	previous value of lagcount, see subclause 5.2.9
prederr	fourth order short term prediction error, see subclause 5.2.5
ptch	Boolean flag indicating the presence of a periodic signal component, see subclause 5.2.9
pvad	energy in the current filtered signal frame, see subclause 5.2.1
rav1	autocorrelation vector obtained from av1, see subclause 5.2.3
rc	the first four unquantized reflection coefficients calculated in the speech encoder (GSM 06.20 (ETS 300 581-2) [2])
rvad	autocorrelation vector of the adaptive filter predictor values, see subclause 5.2.6
smallag	difference between consecutive lag values, see subclause 5.2.9
stat	Boolean flag indicating that the frequency spectrum of the input signal is stationary, see subclause 5.2.4
thvad	adaptive primary VAD threshold, see subclause 5.2.6
tone	Boolean flag indicating the presence of an information tone, see subclause 5.2.5
vadflag	Boolean VAD decision with hangover included, see subclause 5.2.8
veryoldlagcount	previous value of oldlagcount, see subclause 5.2.9
vvad	Boolean VAD decision before hangover, see subclause 5.2.7

3.2.2 Constants

adp	number of frames of hangover for secondary VAD, see subclause 5.2.6
burstconst	minimum length of speech burst to which hangover is added, see subclause 5.2.8
dec	determines rate of decrease in adaptive threshold, see subclause 5.2.6
fac	determines steady state adaptive threshold, see subclause 5.2.6
frames	number of frames over which av0 and av1 are calculated, see subclause 5.2.2
freqth	threshold for pole frequency decision, see subclause 5.2.5
hangconst	number of frames of hangover for primary VAD, see subclause 5.2.8
inc	determines rate of increase in adaptive threshold, see subclause 5.2.6
lthresh	lag difference threshold for periodicity decision, see subclause 5.2.9
margin	determines upper limit for adaptive threshold, see subclause 5.2.6
nthresh	frame count threshold for periodicity decision, see subclause 5.2.9
plev	lower limit for adaptive threshold, see subclause 5.2.6
predth	threshold for short term prediction error, see subclause 5.2.5
pth	energy threshold, see subclause 5.2.6
thresh	decision threshold for evaluation of stat flag, subclause 5.2.4

3.2.3 Functions

+	addition
-	subtraction
*	multiplication
/	division
| x |	absolute value of x
AND	Boolean AND
OR	Boolean OR
$\prod_{i=a}^{b} x(i)$	the product of the series x(i) for i=a to b
$\sum_{i=a}^{b} x(i)$	the sum of the series x(i) for i=a to b

3.3 Abbreviations

ACF	Autocorrelation Function
AFLAT	Autocorrelation Fixed point LAttice Technique
ANSI	American National Standards Institute
DTX	Discontinuous Transmission
LTP	Long Term Predictor
TX	Transmission
VAD	Voice Activity Detector

For abbreviations not given in this subclause see GSM 01.04 (ETR 100) [1]

4 General

The function of the VAD is to indicate whether each 20 ms frame produced by the speech encoder contains speech or not. The output is a Boolean flag (vadflag) which is used by the Transmit (TX) DTX handler defined in GSM 06.41 (ETS 300 581-5) [4].

This ETS is organised as follows:

Clause 5 describes the principles of operation of the VAD. Clause 6 provides an overview of the computational description of the VAD. The computational details necessary for the fixed point implementation of the VAD algorithm are given in the form of an American National Standards Institute (ANSI) C program contained in GSM 06.06 (ETS 300 581-7) [5].

The verification of the VAD is based on the use of digital test sequences which are described in GSM 06.07 (ETS 300 581-8) [6].

The performance of the VAD algorithm is characterised by the amount of audible speech clipping it introduces and the percentage activity it indicates. The characteristics for the VAD defined in this ETS have been established by extensive testing under a wide range of operating conditions. The results are summarised in annex A.

5 Functional description

The purpose of this clause is to give the reader an understanding of the principles of operation of the VAD, whereas GSM 06.06 (ETS 300 581-7) [5] contains the fixed point computational description of the VAD. In the case of discrepancy between the two descriptions, the description in GSM 06.06 (ETS 300 581-7) [5] will prevail.

5.1 Overview and principles of operation

The function of the VAD is to distinguish between noise with speech present and noise without speech present. This is achieved by comparing the energy of a filtered version of the input signal with a threshold. The presence of speech is indicated whenever the threshold is exceeded.

The detection of speech in mobile environments is difficult due to the low speech/noise ratios which are encountered, particularly in moving vehicles. To increase the probability of detecting speech, the input signal is adaptively filtered (see subclause 5.2.1) to reduce its noise content before the voice activity decision is made (see subclause 5.2.7).

The frequency spectrum and level of the noise may vary within a given environment as well as between different environments. It is therefore necessary to adapt the input filter coefficients and energy threshold at regular intervals as described in subclause 5.2.6.

5.2 Algorithm description

The block diagram of the VAD algorithm is shown in figure 1. The individual blocks are described in the following subclauses. The global variables shown in the block diagram are described in table 1.

Table 1: Description of variables in figure 1

Var	Description
acf	The ACF vector which is calculated in the speech encoder (GSM 06.20 (ETS 300 581-2) [2]).
av0	Averaged ACF vector.
av1	A previous value of av0.
lags	The open loop long term predictor lags for the four speech encoder subframes (GSM 06.20 (ETS 300 581-2) [2]).
ptch	Boolean flag indicating the presence of a periodic signal component.
pvad	Energy in the current filtered signal frame.
rav1	Autocorrelation vector obtained from av1.
rc	The first four unquantized reflection coefficients calculated in the speech encoder (GSM 06.20 (ETS 300 581-2) [2]).
rvad	Autocorrelation vector of the adaptive filter predictor values.
stat	Boolean flag indicating that the frequency spectrum of the input signal is stationary.
thvad	Adaptive primary VAD threshold.
tone	Boolean flag indicating the presence of an information tone.
vadflag	Boolean VAD decision with hangover included.
vvad	Boolean VAD decision before hangover.

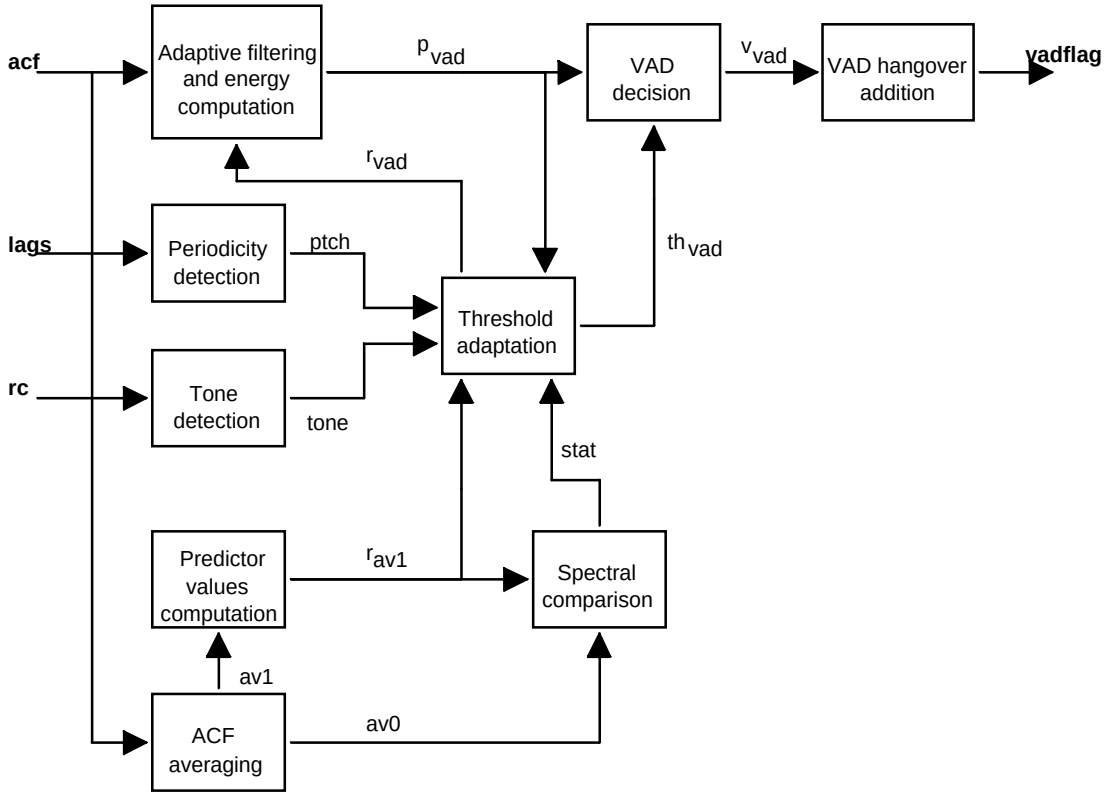

Figure 1: Functional block diagram of the VAD

5.2.1 Adaptive filtering and energy computation

The energy in the current filtered signal frame (pvad) is computed as follows:

$$pvad = rvad[0]*acf[0] + 2*\sum_{i=1}^{8} (rvad[i]*acf[i])$$

This corresponds to performing an 8th order block filtering on the filtered input samples to the speech encoder. This is explained in annex B.

5.2.2 ACF averaging

Spectral characteristics of the input signal have to be obtained using blocks that are larger than one 20 ms frame. This is done by averaging the ACF (autocorrelation function) vectors for several consecutive frames. The averaging is given by the following equations:

$$av0\{n\}[i] = \sum_{j=0}^{frames-1} (acf\{n-j\}[i]) \quad ; \quad i = 0..8$$

$$av1\{n\}[i] = av0\{n-frames\}[i] \quad ; \quad i = 0..8$$

where (n) represents the current frame, (n-1) represents the previous frame etc. The values of the constants and initial variable values are given in table 2.

Table 2: Constants and variables for ACF averaging

Constant	Value	Variable	Initial value
frames	4	previous ACF's, av0 & av1	All set to 0

5.2.3 Predictor values computation

The filter predictor values aav1 are obtained from the autocorrelation values av1 according to the equation:

$$a = R^{-1}p$$

where:

$$R = \begin{bmatrix} av1[0] & av1[1] & av1[2] & av1[3] & av1[4] & av1[5] & av1[6] & av1[7] \\ av1[1] & av1[0] & av1[1] & av1[2] & av1[3] & av1[4] & av1[5] & av1[6] \\ av1[2] & av1[1] & av1[0] & av1[1] & av1[2] & av1[3] & av1[4] & av1[5] \\ av1[3] & av1[2] & av1[1] & av1[0] & av1[1] & av1[2] & av1[3] & av1[4] \\ av1[4] & av1[3] & av1[2] & av1[1] & av1[0] & av1[1] & av1[2] & av1[3] \\ av1[5] & av1[4] & av1[3] & av1[2] & av1[1] & av1[0] & av1[1] & av1[2] \\ av1[6] & av1[5] & av1[4] & av1[3] & av1[2] & av1[1] & av1[0] & av1[1] \\ av1[7] & av1[6] & av1[5] & av1[4] & av1[3] & av1[2] & av1[1] & av1[0] \end{bmatrix}$$

and:

$$p = \begin{bmatrix} av1[1] \\ av1[2] \\ av1[3] \\ av1[4] \\ av1[5] \\ av1[6] \\ av1[7] \\ av1[8] \end{bmatrix} \qquad a = \begin{bmatrix} aav1[1] \\ aav1[2] \\ aav1[3] \\ aav1[4] \\ aav1[5] \\ aav1[6] \\ aav1[7] \\ aav1[8] \end{bmatrix}$$

$$aav1[0] = -1$$

av1 is used in preference to av0 as the latter may contain speech. The autocorrelated predictor values rav1 are then obtained:

```
          8-i
rav1[i] = SUM (aav1[k]*aav1[k+i])     ; i = 0..8
          k=0
```

5.2.4 Spectral comparison

The spectra represented by the autocorrelated predictor values rav1 and the averaged autocorrelation values av0 are compared using the distortion measure (dm), defined below. This measure is used to produce a Boolean value stat every 20 ms, as shown in the following equations:

$$dm = (rav1[0]*av0[0] + 2*\sum_{i=1}^{8} (rav1[i]*av0[i]))/av0[0]$$

$$difference = | dm - lastdm |$$

$$lastdm = dm$$

$$stat = (difference < thresh)$$

The values of the constants and initial variable values are given in table 3.

Table 3: Constants and variables for spectral comparison

Constant	Value	Variable	Initial value
thresh	0,068	lastdm	0

5.2.5 Information tone detection

Information tones and noise can be classified by inspecting the short term prediction gain, information tones resulting in a higher prediction gain than noise. Tones can therefore be detected by comparing the prediction gain to a fixed threshold. By limiting the prediction gain calculation to a fourth order analysis, information signals consisting of one or two tones can be detected whilst minimising the prediction gain for noise.

The prediction gain decision is implemented by comparing the normalised short term prediction error with the short term prediction error threshold (predth). This measure is used to produce a Boolean value, tone, every 20 ms. The signal is classified as a tone if the prediction error is less than predth. This is equivalent to a prediction gain threshold of 13.5 dB.

Vehicle noise can contain strong resonances at low frequencies, resulting in a high prediction gain. A further test is therefore made to determine the pole frequency of a second order analysis of the signal frame. The signal is classified as noise if the frequency of the pole is less than 385 Hz.

The algorithm for evaluating the Boolean tone flag is as follows:

```
tone = false

den = a[1]*a[1]
num = 4*a[2] - a[1]*a[1]

if (num <= 0)
  return

if ((a[1] < 0) AND (num/den < freqth))
  return
           4
prederr = MULT (1 - rc[i]*rc[i])
          i=1

if (prederr < predth)
  tone = true

return
```

rc[1..4] are the first four unquantized reflection coefficients obtained from the speech encoder short term predictor. The coefficients a[0..2] are transversal filter coefficients calculated from rc[1..2] using the step up routine described in subclause 6.3.3. The pole frequency calculation is described in annex C.

The values of the constants are given in table 4.

Table 4: Constants for information tone detection

Constant	Value
freqth	0,0973
predth	0,0447

5.2.6 Threshold adaptation

A check is made every 20 ms to determine whether the adaptive primary VAD threshold, (thvad) should be changed. This adaptation is carried out according to the flow chart in figure 2. The values of the constants and initial variable values are given in table 5.

Adaptation of thvad takes place in two different situations:

In the first case, the decision threshold (thvad) is set to the lower limit for the adaptive threshold (plev) if the input signal frame energy (acf[0]) is less than the energy threshold (pth). The autocorrelation vector of the adaptive filter predictor values (rvad) remains unchanged.

In the second case, thvad and rvad are adapted if there is a low probability that speech or information tones are present. This occurs when the following conditions are met:

a) The frequency spectrum of the input signal is stationary (subclause 5.2.4).

b) The signal does not contain a periodic component (subclause 5.2.9).

c) Information tones are not present (subclause 5.2.5).

The autocorrelation vector of the adaptive filter predictor values (rvad) is updated with the rav1 values. The step size by which thvad is adapted is not constant but a proportion of the current value and its rate of increase or decrease is determined by constants inc and dec respectively.

The adaptation begins by experimentally multiplying thvad by a factor of (1-1/dec). If thvad is now higher than or equal to pvad times, the steady state adaptive threshold constant (fac), then thvad needed to be decreased and it is left at this new lower level. If, on the other hand, thvad is less than pvad times fac, then it either needs to be increased or kept constant. In this case, it is multiplied by a factor of (1+1/inc) or set to pvad times fac whichever yields the lower value. thvad is never allowed to be greater than pvad+upper adaptive threshold limit (margin).

Table 5: Constants and variables threshold adaptation

Constant	Value	Variable	Initial value
pth	210000	margin	112000000
plev	560000	adaptcount	0
fac	2,55	thvad	1400000
adp	8	rvad[0]	6
inc	16	rvad[1] to	All 0
dec	32	rvad[8]	

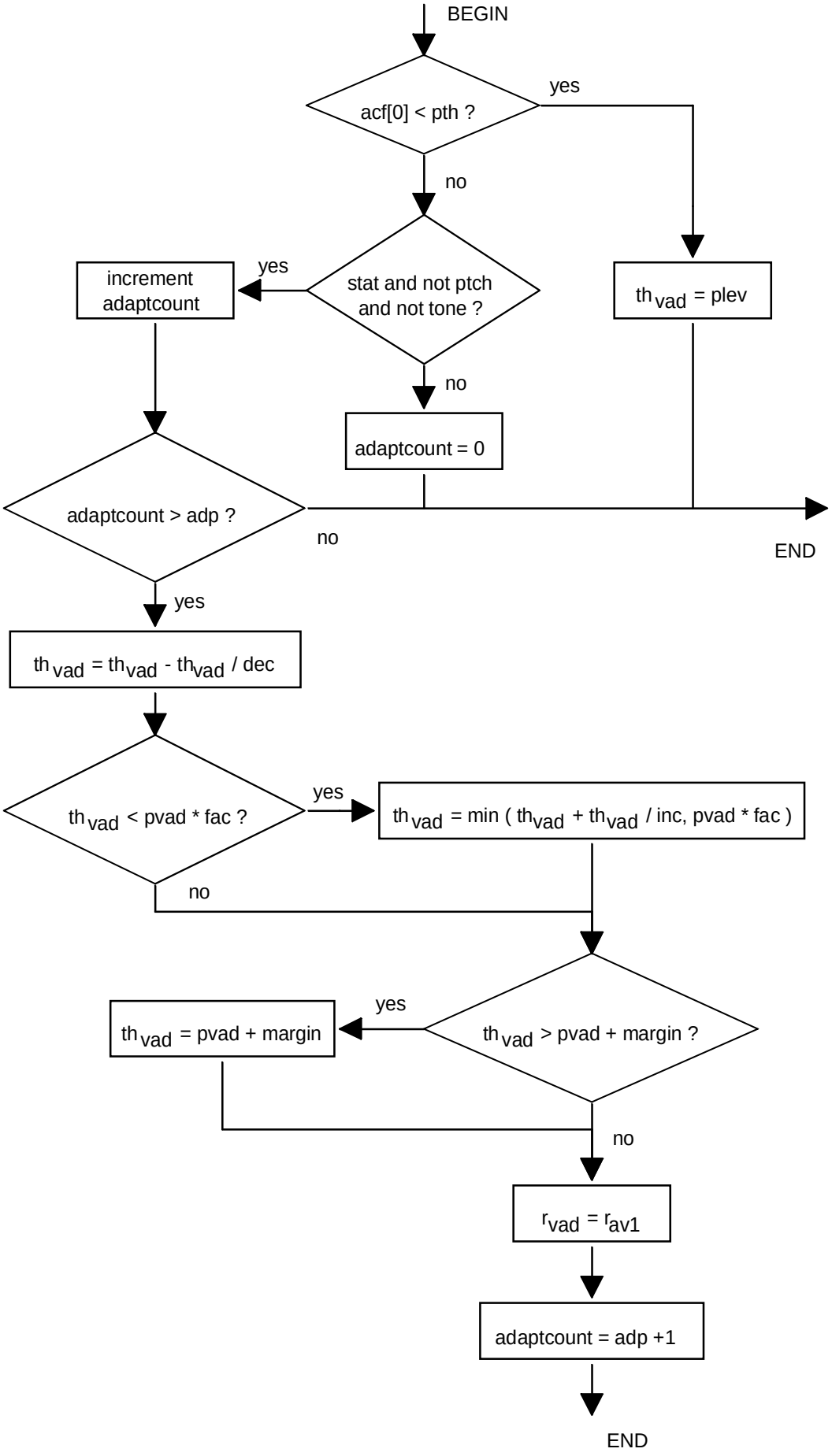

Figure 2: Flow diagram for threshold adaptation

5.2.7 VAD decision

Prior to hangover the Boolean VAD decision is defined as:

```
vvad = (pvad > thvad)
```

5.2.8 VAD hangover addition

VAD hangover is only added to bursts of speech greater than or equal to burstcount blocks. The Boolean variable vadflag indicates the decision of the VAD with hangover included. The values of the constants and initial variable values are given in table 6. The hangover algorithm is as follows:

```
if (vvad)
     increment(burstcount)
else
     burstcount = 0

if (burstcount >= burstconst)
     {
     hangcount = hangconst
     burstcount = burstconst
     }

vadflag = (vvad OR (hangcount >= 0))

if (hangcount >= 0)
     decrement(hangcount)
```

Table 6: Constants and variables for VAD hangover addition

Constant	Value	Variable	Initial value
burstconst	3	burstcount	0
hangconst	5	hangcount	-1

5.2.9 Periodicity detection

thvad and rvad are updated when the frequency spectrum of the input signal is stationary. However, vowel sounds also have a stationary frequency spectrum. The Boolean variable ptch indicates the presence of a periodic signal component and prevents adaptation of thvad and rvad.ptch is updated every 20 ms and is true when periodicity (a vowel sound) is detected. The periodicity detector identifies the vowel sounds by comparing consecutive Long Term Predictor (LTP) lag values lags[1..4] which are obtained every sub frame from the speech codec defined in GSM 06.20 (ETS 300 581-2) [2]. Cases in which one lag value is a factor of the other are catered for. However, consecutive lags with a ratio which is either non-integer or greater than 3 are not classified as periodic.

```
lagcount = 0

for ( j=1; j<=4; j++ )
     {
     smallag=maximum(lags[j],lags[j-1])
     for ( i=1; i<=3; i++ )
          if (smallag >= minimum(lags[i], lags[i-1]))
               smallag = smallag - minimum(lags[i], lags[i-1])
     if (minimum(smallag,minimum(lags[j],lags[j-1])-smallag)<lthresh)
          increment(lagcount)
     }

veryoldlagcount = oldlagcount

oldlagcount = lagcount

ptch = (oldlagcount + veryoldlagcount >= nthresh)
```

The values of constants and initial values are given in table 7. lags[0] = lags[4] of the previous frame.

ptch is calculated after the VAD decision and when the current LTP lag values lags[1..4] are available. This reduces the delay of the VAD decision.

Table 7: Constants and variables for periodicity detection

Constant	Value	Variable	Initial value
lthresh	2	ptch	1
nthresh	7	oldlagcount	0
		veryoldlagcount	0
		lags[0]	21

6 Computational description overview

The computational details necessary for the fixed point implementation of the speech transcoding and DTX functions are given in the form of an American National Standards Institute (ANSI) C program contained in GSM 06.06 (ETS 300 581-7) [5]. This clause provides an overview of the modules which describe the computation of the VAD algorithm.

6.1 VAD modules

The computational description of the VAD is divided into three ANSI C modules. These modules are:

- vad_reset

- vad_algorithm

- periodicity_update

The vad_reset module sets the VAD variables to their initial values.

The vad_algorithm module is divided into nine sub-modules which correspond to the blocks of figure 1 in the high level description of the VAD algorithm. The vad_algorithm module can be called as soon as the acf[0..8] and rc[1..4] variables are known. This means that the VAD computation can take place after the Autocorrelation Fixed point LAttice Technique (AFLAT) routine in the speech encoder (GSM 06.20 (ETS 300 581-2) [2]). The vad_algorithm module also requires the value of the ptch variable calculated in the previous frame.

The ptch variable is calculated by the periodicity_update module from the lags[1..4] variable. The individual lag values are calculated for each subframe in the LTP routine of the speech encoder (GSM 06.20 (ETS 300 581-2) [2]). The periodicity_update module is called after the current 20 ms signal frame has been encoded.

6.2 Pseudo-floating point arithmetic

All the arithmetic operations follow the precision and format used in the computational description of the speech codec in GSM 06.06 (ETS 300 581-7) [5]. To increase the precision within the fixed point implementation, a pseudo-floating point representation of some variables is used. This applies to the following variables (and related constants) of the VAD algorithm:

- pvad: Energy of filtered signal;
- thvad: Threshold of the VAD decision;
- acf0: Energy of the input signal.

For the representation of these variables, two 16-bit integers are needed

- one for the exponent (e_pvad, e_thvad, e_acf0);
- one for the mantissa (m_pvad, m_thvad, m_acf0).

The value e_pvad represents the lowest power of 2 just greater or equal to the actual value of pvad, and the m_pvad value represents an integer which is always greater than or equal to 16384 (normalised mantissa). It means that the pvad value is equal to

$$pvad = 2^{e\_pvad} * (m\_pvad / 32768)$$

This scheme provides a large dynamic range for the pvad value and always keeps a precision of 16 bits. All the comparisons are easy to make by comparing the exponents of two variables, and the VAD algorithm needs only one pseudo floating point addition and multiplication. All the computations related to the pseudo-floating point variables require simple 16 or 32-bit arithmetic operations defined in the detailed description of the speech codec.

Some constants, represented by a floating point format, are needed and symbolic names (in capital letters) for their exponent and mantissa are used; table 8 lists all these constants with the associated symbolic names and their numerical constant values.

Table 8: List of floating point constants

Constant	Exponent	Mantissa
pth	E_PTH = 18	M_PTH = 26250
margin	E_MARGIN = 27	M_MARGIN = 27343
plev	E_PLEV = 20	M_PLEV = 17500

Annex A (informative): VAD performance

In the optimisation of a VAD, a trade-off has to be made between speech clipping, which reduces the subjective performance of the system, and the mean channel activity factor. The benefit of DTX is increased as the activity factor is reduced. However, in general, a reduction of the activity factor will be associated with a greater risk of audible speech clipping.

In the optimisation process, emphasis has been placed on avoiding unnecessary speech clipping. However, it has been found that a VAD with virtually no audible clipping would result in a high activity and little DTX advantage. The VAD specified in this ETS introduces audible and possibly objectionable clipping in certain cases, mainly for low input levels and low signal to noise ratios.

An indication of the mean channel activity in DTX mode is given in table A.1. The figure quoted is the average calculated over a large number of conversations covering factors such as different talkers, noise characteristics and locations. It should be noted that the actual activity of a particular talker in a specific conversation may vary considerably from the figure given in the table. This is due to both talker behaviour and the level dependency of the VAD (the channel activity has been found to decrease by about 0.5% per dB of level reduction). However, as mentioned above, a decreased speech input level increases the risk of objectionable clipping.

Table A.1: Mean channel activity factor in DTX mode

Channel activity factor
60%

Annex B (informative): Simplified block filtering operation

Consider an 8th order transversal filter with filter coefficients a[0..8], through which a signal is being passed, the output of the filter being:

$$s'n = -\sum_{i=0}^{8} (a[i]*s[n-i]) \qquad (1)$$

If we apply block filtering over 20 ms frames, then this equation becomes:

$$s'n = -\sum_{i=0}^{\min(8,n)} (a[i]*s[n-i]) \quad ; n = 0..167 \quad ; 0 <= n <= 167 \qquad (2)$$

If the energy of the filtered signal is then obtained for every 20 ms frame, the equation for this is:

$$pvad = \sum_{n=0}^{167} \left(-\sum_{i=0}^{\min(8,n)} (a[i]*s[n-i]) \right)^2 \quad ; 0 <= n-i <= 159 \qquad (3)$$

We know that:

$$acf[i] = \sum_{n=i}^{159} (s[n]*s[n-i]) \quad ; i = 0..8 \quad ; 0 <= n-i <= 159 \qquad (4)$$

If equation (3) is expanded and acf[0..8] are substituted for s[n] then we arrive at the equations:

$$pvad = r[0]*acf[0] + 2*\sum_{i=1}^{8} (r[i]*acf[i]) \qquad (5)$$

Where:

$$r[i] = \sum_{k=0}^{8-i} (a[k]*a[k+i]) \quad ; i = 0..8 \qquad (6)$$

Annex C (informative): Pole frequency calculation

This annex describes the algorithm used to determine whether the pole frequency for a second order analysis of the signal frame is less than 385 Hz.

The filter coefficients for a second order synthesis filter are calculated from the first two unquantized reflection coefficients rc[1..2] obtained from the speech encoder. This is done using the step up routine described in subclause 6.3.3. If the filter coefficients a[0..2] are defined such that the synthesis filter response is given by:

$$H(z) = 1/(a[0] + a[1]z^{-1} + a[2]z^{-2}) \quad (1)$$

Then the positions of the poles in the Z-plane are given by the solutions to the following quadratic:

$$a[0]z^2 + a[1]z + a[2] = 0, \quad a[0] = 1 \quad (2)$$

The positions of the poles, z, are therefore:

$$z = re \pm j*sqrt(im), \qquad j^2 = -1 \quad (3)$$

where:

$$re = - a[1] / 2 \quad (4)$$

$$im = (4*a[2] - a[1]^2)/4 \quad (5)$$

If im is negative then the poles lie on the real axis of the Z-plane and the signal is not a tone and the algorithm terminates. If re is negative then the poles lie in the left hand side of the Z-plane and the frequency is greater than 2000 Hz and the prediction error test can be performed.

If im is positive and re is positive then the poles are complex and lie in the right hand side of the Z-plane and the frequency in Hz is related to re and im by the expression:

$$freq = \arctan(sqrt(im)/re)*4000/pi \quad (6)$$

Having ensured that both im and re are positive the test for a pole frequency less than 385 Hz can be derived by substituting equations 4 and 5 into equation 6 and re-arranging:

$$(4*a[2] - a[1]^2)/a[1]^2 < \tan^2(pi*385/4000) \quad (7)$$

or

$$(4*a[2] - a[1]^2)/a[1]^2 < 0.0973 \quad (8)$$

If this test is true then the signal is not a tone and the algorithm terminates, otherwise the prediction error test is performed.

History

Document history			
March 1995	Public Enquiry	PE 80:	1995-03-06 to 1995-06-30
August 1995	Vote	V 86: extended:	1995-08-21 to 1995-10-13 1995-08-21 to 1995-10-27
November 1995	First Edition		

ISBN 2-7437-0398-9 - Partie 6
Dépôt légal : Novembre 1995