ETSI TS 126 091 V9.0.0 (2010-01)

Technical Specification

**Digital cellular telecommunications system (Phase 2+);
Universal Mobile Telecommunications System (UMTS);
LTE;
Mandatory Speech Codec speech processing functions;
Adaptive Multi-Rate (AMR) speech codec;
Error concealment of lost frames
(3GPP TS 26.091 version 9.0.0 Release 9)**

Reference

RTS/TSGS-0426091v900

Keywords

GSM, LTE, UMTS

ETSI

650 Route des Lucioles
F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - NAF 742 C
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° 7803/88

Important notice

Copyright Notification

Intellectual Property Rights

IPRs essential or potentially essential to the present document may have been declared to ETSI. The information pertaining to these essential IPRs, if any, is publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: *"Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards"*, which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (http://webapp.etsi.org/IPR/home.asp).

Pursuant to the ETSI IPR Policy, no investigation, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Foreword

This Technical Specification (TS) has been produced by ETSI 3rd Generation Partnership Project (3GPP).

The present document may refer to technical specifications or reports using their 3GPP identities, UMTS identities or GSM identities. These should be interpreted as being references to the corresponding ETSI deliverables.

The cross reference between GSM, UMTS, 3GPP and ETSI identities can be found under http://webapp.etsi.org/key/queryform.asp.

Contents

Intellectual Property Rights 2
Foreword 2
Foreword 4
1 Scope 5
2 References 5
3 Definitions and abbreviations 5
3.1 Definitions 5
3.2 Abbreviations 5
4 General 6
5 Requirements 6
5.1 Error detection 6
5.2 Lost speech frames 6
5.3 First lost SID frame 6
5.4 Subsequent lost SID frames 6
6 Example ECU/BFH Solution 1 6
6.1 State Machine 7
6.2 Assumed Active Speech Frame Error Concealment Unit Actions 8
6.2.1 BFI = 0, prevBFI = 0, State = 0 8
6.2.2 BFI = 0, prevBFI = 1, State = 0 or 5 8
6.2.3 BFI = 1, prevBFI = 0 or 1, State = 1...6 9
6.2.3.1 LTP-lag update 9
6.2.3.2 Innovation sequence 9
6.3 Assumed Non-Active Speech Signal Error Concealment Unit Actions 10
6.3.1 General 10
6.3.2 Detectors 10
6.3.2.1 Background detector 10
6.3.2.2 Voicing detector 10
6.3.3 Background ECU Actions 10
6.4 Substitution and muting of lost SID frames 10
7 Example ECU/BFH Solution 2 11
7.1 State Machine 11
7.2 Substitution and muting of lost speech frames 11
7.2.1 BFI = 0, prevBFI = 0, State = 0 11
7.2.2 BFI = 0, prevBFI = 1, State = 0 or 5 11
7.2.3 BFI = 1, prevBFI = 0 or 1, State = 1...6 12
7.2.3.1 LTP-lag update 12
7.2.4 Innovation sequence 12
7.3 Substitution and muting of lost SID frames 12
Annex A (informative): Change history 13
History 14

Foreword

This Technical Specification has been produced by the 3rd Generation Partnership Project (3GPP).

The contents of the present document are subject to continuing work within the TSG and may change following formal TSG approval. Should the TSG modify the contents of the present document, it will be re-released by the TSG with an identifying change of release date and an increase in version number as follows:

Version x.y.z

where:

- x the first digit:
  - 1 presented to TSG for information;
  - 2 presented to TSG for approval;
  - 3 or greater indicates TSG approved document under change control.
- y the second digit is incremented for all changes of substance, i.e. technical enhancements, corrections, updates, etc.
- z the third digit is incremented when editorial only changes have been incorporated in the document.

1 Scope

The present document defines an error concealment procedure, also termed frame substitution and muting procedure, which shall be used by the AMR speech codec receiving end when one or more lost speech or lost Silence Descriptor (SID) frames are received.

The requirements of the present document are mandatory for implementation in all networks and User Equipment (UE)s capable of supporting the AMR speech codec. It is not mandatory to follow the bit exact implementation outlined in the present document and the corresponding C source code.

2 References

The following documents contain provisions which, through reference in this text, constitute provisions of the present document.

- References are either specific (identified by date of publication, edition number, version number, etc.) or non-specific.
- For a specific reference, subsequent revisions do not apply.
- For a non-specific reference, the latest version applies. In the case of a reference to a 3GPP document (including a GSM document), a non-specific reference implicitly refers to the latest version of that document *in the same Release as the present document*.

[1] 3GPP TS 26.102: "AMR Speech Codec; Interface to Iu snd Uu".

[2] 3GPP TS 26.090: "Transcoding functions".

[3] 3GPP TS 26.093: "Source Controlled Rate operation".

[4] 3GPP TS 26.101: "Frame Structure".

3 Definitions and abbreviations

3.1 Definitions

For the purposes of the present document, the following terms and definitions apply:

N-point median operation: consists of sorting the N elements belonging to the set for which the median operation is to be performed in an ascending order according to their values, and selecting the (int (N/2) + 1) -th largest value of the sorted set as the median value

Further definitions of terms used in the present document can be found in the references.

3.2 Abbreviations

For the purposes of the present document, the following abbreviations apply:

AN	Access Network
BFH	Bad Frame Handling
BFI	Bad Frame Indication from AN
BSI_netw	Bad Sub-block Indication obtained from AN interface CRC checks
CRC	Cyclic Redundancy Check
ECU	Error Concealment Unit
medianN	N-point median operation
PDFI	Potentially Degraded Frame Indication

prevBFI	Bad Frame Indication of previous frame
RX	Receive
SCR	Source Controlled Rate (operation)
SID	Silence Descriptor frame (Background descriptor)

4 General

The purpose of the error concealment procedure is to conceal the effect of lost AMR speech frames. The purpose of muting the output in the case of several lost frames is to indicate the breakdown of the channel to the user and to avoid generating possible annoying sounds as a result from the error concealment procedure.

The network shall indicate lost speech or lost SID frames by setting the RX_TYPE values [3] to SPEECH_BAD or SID_BAD. If these flags are set, the speech decoder shall perform parameter substitution to conceal errors.

The network should also indicate potentially degraded frames using the flag RX_TYPE value SPEECH_PROBABLY_DEGRADED. This flag may be derived from channel quality indicators. It may be used by the speech decoder selectively depending on the estimated signal type.

The example solutions provided in paragraphs 6 and 7 apply only to bad frame handling on a complete speech frame basis. Sub-frame based error concealment may be derived using similar methods.

5 Requirements

5.1 Error detection

If the most sensitive bits of the AMR speech data (class A in [4]) are received in error, the network shall indicate RX_TYPE = SPEECH_BAD in which case the BFI flag is set. If a SID frame is received in error, the network shall indicate RX_TYPE = SID_BAD in which case the BFI flag is also set. The RX_TYPE = SPEECH_PROBABLY_DEGRADED flag should be set appropriately using quality information from the channel decoder, in which case the PDFI flag is set.

5.2 Lost speech frames

Normal decoding of lost speech frames would result in very unpleasant noise effects. In order to improve the subjective quality, lost speech frames shall be substituted with either a repetition or an extrapolation of the previous good speech frame(s). This substitution is done so that it gradually will decrease the output level, resulting in silence at the output. Clauses 6, and 7 provide example solutions.

5.3 First lost SID frame

A lost SID frame shall be substituted by using the SID information from earlier received valid SID frames and the procedure for valid SID frames be applied as described in [3].

5.4 Subsequent lost SID frames

For many subsequent lost SID frames, a muting technique shall be applied to the comfort noise that will gradually decrease the output level. For subsequent lost SID frames, the muting of the output shall be maintained. Clauses 6 and 7 provide example solutions.

6 Example ECU/BFH Solution 1

The C code of the following example is embedded in the bit exact software of the codec. In the code the ECU is designed to allow subframe-by-subframe synthesis, thereby reducing the speech synthesis delay to a minimum.

6.1 State Machine

This example solution for substitution and muting is based on a state machine with seven states (Figure 1).

The system starts in state 0. Each time a bad frame is detected, the state counter is incremented by one and is saturated when it reaches 6. Each time a good speech frame is detected, the state counter is reset to zero, except when we are in state 6, where we set the state counter to 5. The state indicates the quality of the channel: the larger the value of the state counter, the worse the channel quality is. The control flow of the state machine can be described by the following C code (**BFI** = bad frame indicator, **State** = state variable):

```
if(BFI != 0 )
    State = State + 1;
else if(State == 6)
    State = 5;
else
    State = 0;
if(State > 6 )
    State = 6;
```

In addition to this state machine, the **Bad Frame Flag** from the previous frame is checked (**prevBFI**). The processing depends on the value of the **State**-variable. In states 0 and 5, the processing depends also on the two flags **BFI** and **prevBFI**.

The procedure can be described as follows:

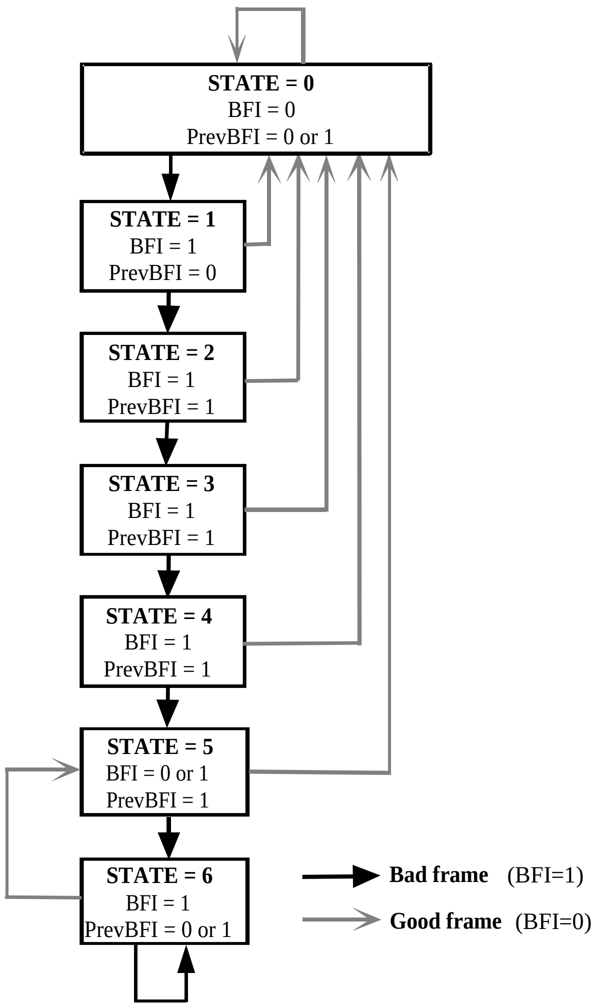

Figure 1: State machine for controlling the bad frame substitution

6.2 Assumed Active Speech Frame Error Concealment Unit Actions

6.2.1 BFI = 0, prevBFI = 0, State = 0

No error is detected in the received or in the previous received speech frame. The received speech parameters are used in the normal way in the speech synthesis. The current frame of speech parameters is saved.

6.2.2 BFI = 0, prevBFI = 1, State = 0 or 5

No error is detected in the received speech frame, but the previous received speech frame was bad. The LTP gain and fixed codebook gain are limited below the values used for the last received good

subframe: $$g^p = \begin{cases} g^p, & g^p \le g^p(-1) \\ g^p(-1), & g^p > g^p(-1) \end{cases} \qquad (1)$$

where g^p = current decoded LTP gain, $g^p(-1)$ = LTP gain used for the last good subframe (BFI = 0), and

$$g^c = \begin{cases} g^c, & g^c \le g^c(-1) \\ g^c(-1), & g^c > g^c(-1) \end{cases} \quad (2)$$

where g^c = current decoded fixed codebook gain and $g^c(-1)$ = fixed codebook gain used for the last good subframe (BFI = 0).

The rest of the received speech parameters are used normally in the speech synthesis. The current frame of speech parameters is saved.

6.2.3 BFI = 1, prevBFI = 0 or 1, State = 1...6

An error is detected in the received speech frame and the substitution and muting procedure is started. The LTP gain and fixed codebook gain are replaced by attenuated values from the previous subframes:

$$g^p = \begin{cases} P(state)\ g^p(-1), & g^p(-1) \le median5(g^p(-1),\ldots,g^p(-5)) \\ P(state)\ median5(g^p(-1),\ldots,g^p(-5)), & g^p(-1) > median5(g^p(-1),\ldots,g^p(-5)) \end{cases} \quad (3)$$

where g^p = current decoded LTP gain, $g^p(-1),\ldots,g^p(-n)$ = LTP gains used for the last n subframes, *median5()* = 5-point median operation, *P(state)* = attenuation factor (*P(1)* = 0.98, *P(2)* = 0.98, *P(3)* = 0.8, *P(4)* = 0.3, *P(5)* = 0.2, *P(6)* = 0.2), *state* = state number, and

$$g^c = \begin{cases} C(state)\ g^c(-1), & g^c(-1) \le median5(g^c(-1),\ldots,g^c(-5)) \\ C(state)\ median5(g^c(-1),\ldots,g^c(-5)), & g^c(-1) > median5(g^c(-1),\ldots,g^c(-5)) \end{cases} \quad (4)$$

where g^c = current decoded fixed codebook gain, $g^c(-1),\ldots,g^c(-n)$ = fixed codebook gains used for the last n subframes, *median5()* = 5-point median operation, *C(state)* = attenuation factor (*C(1)* = 0.98, *C(2)* = 0.98, *C(3)* = 0.98, *C(4)* = 0.98, *C(5)* = 0.98, *C(6)* = 0.7), and *state* = state number.

The higher the state value is, the more the gains are attenuated. Also the memory of the predictive fixed codebook gain is updated by using the average value of the past four values in the memory:

$$ener(0) = \frac{1}{4}\sum_{i=1}^{4} ener(-i) \quad (5)$$

The past LSFs are shifted towards their mean:

$$lsf\_q1(i) = lsf\_q2(i) = \alpha\ past\_lsf\_q(i) + (1-\alpha)mean\_lsf(i), \quad i = 0\ldots9 \quad (6)$$

where α = 0.95, *lsf_q1* and *lsf_q2* are two sets of LSF-vectors for current frame, *past_lsf_q* is *lsf_q2* from the previous frame, and *mean_lsf* is the average LSF-vector. Note that two sets of LSFs are available only in the 12.2 mode.

6.2.3.1 LTP-lag update

The LTP-lag values are replaced by the past value from the 4th subframe of the previous frame (12.2 mode) or slightly modified values based on the last correctly received value (all other modes).

6.2.3.2 Innovation sequence

The received fixed codebook innovation pulses from the erroneous frame are used in the state in which they were received when corrupted data are received . In the case when no data were received random fixed codebook indices should be employed.

6.3 Assumed Non-Active Speech Signal Error Concealment Unit Actions

6.3.1 General

The Non-Active Speech ECU is used to reduce the negative impact of amplitude variations and tonal artefacts when using the conventional Active Speech ECU in non-voiced signals such as background noise and unvoiced speech. The background ECU actions are only used for the lower rate Speech Coding modes.

The Non-Active Speech ECU actions are done as postprocessing actions of the Active Speech ECU, actions thus ensuring that the Active Speech ECU states are continuously updated. This will guarantee instant and seamless switching to the Active Speech ECU. The detectors and state updates have to be running continuously for all speech coding modes to avoid switching problems.

Only the differences to the Active Speech ECU are stated below.

6.3.2 Detectors

6.3.2.1 Background detector

An energy level and energy change detector is used to monitor the signal. If the signal is considered to contain background noise and only shows minor energy level changes, a flag is set. The resulting indicator is the **inBackgroundNoise** flag which indicates the signal state of the previous frame.

6.3.2.2 Voicing detector

The received LTP gain is monitored and used to prevent the use of the background ECU actions in possibly voiced segments. A median filtered LTP gain value with a varying filter memory length is thresholded to provide the correct voicing decision. Additionally, a counter **voicedHangover** is used to monitor the time since a frame was presumably voiced.

6.3.3 Background ECU Actions

The BFI, and DFI indications are used together with the flag **inBackgroundNoise** and the counter **voicedHangover** to adjust the LTP part and the innovation part of the excitation. The actions are only taken if the previous frame has been classified as background noise and sufficient time has passed since the last voiced frame was detected.

The background ECU actions are: energy control of the excitation signal, relaxed LTP lag control, stronger limitation of the LTP gain, adjusted adaptation of the Gain-Contour-Smoothing algorithm and modified adaptation of the Anti-Sparseness Procedure.

6.4 Substitution and muting of lost SID frames

In the speech decoder a single frame classified as SID_BAD shall be substituted by the last valid SID frame information and the procedure for valid SID frames be applied. If the time between SID information updates (updates are specified by SID_UPDATE arrivals and occasionally by SID_FIRST arrivals see 06.92) is greater than one second this shall lead to attenuation.

7 Example ECU/BFH Solution 2

This is an alternative example solution which is a simplified version of Example ECU/BFH Solution 1.

7.1 State Machine

This example solution for substitution and muting is based on a state machine with seven states (Figure 1, same state machine as in Example 1).

The system starts in state 0. Each time a bad frame is detected, the state counter is incremented by one and is saturated when it reaches 6. Each time a good speech frame is detected, the state counter is reset to zero, except when we are in state 6, where we set the state counter to 5. The state indicates the quality of the channel: the larger the state counter, the worse the channel quality is. The control flow of the state machine can be described by the following C code (**BFI** = bad frame indicator, **State** = state variable):

```
if(BFI != 0 )
     State = State + 1;
else if(State == 6)
     State = 5;
else
     State = 0;
if(State > 6 )
     State = 6;
```

In addition to this state machine, the **Bad Frame Flag** from the previous frame is checked (**prevBFI**). The processing depends on the value of the **State**-variable. In states 0 and 5, the processing depends also on the two flags **BFI** and **prevBFI**.

7.2 Substitution and muting of lost speech frames

7.2.1 BFI = 0, prevBFI = 0, State = 0

No error is detected in the received or in the previous received speech frame. The received speech parameters are used normally in the speech synthesis. The current frame of speech parameters is saved.

7.2.2 BFI = 0, prevBFI = 1, State = 0 or 5

No error is detected in the received speech frame but the previous received speech frame was bad. The LTP gain and fixed codebook gain are limited below the values used for the last received good subframe:

$$g^p = \begin{cases} g^p, & g^p \le g^p(-1) \\ g^p(-1), & g^p > g^p(-1) \end{cases} \tag{7}$$

where g^p = current decoded LTP gain, $g^p(-1)$ = LTP gain used for the last good subframe (BFI = 0), and

$$g^c = \begin{cases} g^c, & g^c \le g^c(-1) \\ g^c(-1), & g^c > g^c(-1) \end{cases} \tag{8}$$

where g^c = current decoded fixed codebook-gain and $g^c(-1)$ = fixed codebook gain used for the last good subframe (BFI = 0).

The rest of the received speech parameters are used normally in the speech synthesis. The current frame of speech parameters is saved.

7.2.3 BFI = 1, prevBFI = 0 or 1, State = 1...6

An error is detected in the received speech frame and the substitution and muting procedure is started. The LTP gain and fixed codebook gain are replaced by attenuated values from the previous subframes:

$$g^p = \begin{cases} P(state)\ g^p(-1), & g^p(-1) \le median5(g^p(-1),\ldots,g^p(-5)) \\ P(state)\ median5(g^p(-1),\ldots,g^p(-5)), & g^p(-1) > median5(g^p(-1),\ldots,g^p(-5)) \end{cases} \quad (9)$$

where g^p = current decoded LTP gain, $g^p(-1),\ldots,g^p(-n)$ = LTP gains used for the last n subframes, *median5()* = 5-point median operation, *P(state)* = attenuation factor (*P(1)* = 0.98, *P(2)* = 0.98, *P(3)* = 0.8, *P(4)* = 0.3, *P(5)* = 0.2, *P(6)* = 0.2), *state* = state number, and

$$g^c = \begin{cases} C(state)\ g^c(-1), & g^c(-1) \le median5(g^c(-1),\ldots,g^c(-5)) \\ C(state)\ median5(g^c(-1),\ldots,g^c(-5)), & g^c(-1) > median5(g^c(-1),\ldots,g^c(-5)) \end{cases} \quad (10)$$

where g^c = current decoded fixed codebook gain, $g^c(-1),\ldots,g^c(-n)$ = fixed codebook gains used for the last n subframes, *median5()* = 5-point median operation, *C(state)* = attenuation factor (*C(1)* = 0.98, *C(2)* = 0.98, *C(3)* = 0.98, *C(4)* = 0.98, *C(5)* = 0.98, *C(6)* = 0.7), and *state* = state number.

The higher the state value is, the more the gains are attenuated. Also the memory of the predictive fixed codebook gain is updated by using the average value of the past four values in the memory:

$$ener(0) = \frac{1}{4}\sum_{i=1}^{4} ener(-i) \quad (11)$$

The past LSFs are used by shifting their values towards their mean:

$$lsf\_q1(i) = lsf\_q2(i) = \alpha\ past\_lsf\_q(i) + (1-\alpha)mean\_lsf(i), \quad i = 0\ldots9 \quad (12)$$

where α = 0.95, *lsf_q1* and *lsf_q2* are two sets of LSF-vectors for current frame, *past_lsf_q* is *lsf_q2* from the previous frame, and *mean_lsf* is the average LSF-vector. Note that two sets of LSFs are available only in the 12.2 mode.

7.2.3.1 LTP-lag update

The LTP-lag values are replaced by the past value from the 4th subframe of the previous frame (12.2 mode) or slightly modified values based on the last correctly received value (all other modes).

7.2.4 Innovation sequence

The received fixed codebook innovation pulses from the erroneous frame are used in the state in which they were received when corrupted data are received. In the case when no data were received random fixed codebook indices should be employed.

7.3 Substitution and muting of lost SID frames

In the speech decoder a single frame classified as SID_BAD shall be substituted by the last valid SID frame information and the procedure for valid SID frames be applied. If the time between SID information updates (updates are specified by SID_UPDATE arrivals and occasionally by SID_FIRST arrivals) is greater than one second this shall lead to attenuation.

Annex A (informative): Change history

Tdoc	SPEC	CR	RE	VER	SUBJECT	CAT	NEW
SP-99570	26.091	A001		3.0.1	Use of random excitation when RX_NODATA and not in DTX	F	3.1.0

Change history							
Date	**TSG #**	**TSG Doc.**	**CR**	**Rev**	**Subject/Comment**	**Old**	**New**
03-2001	11				Version for Release 4		4.0.0
06-2002	16				Version for Release 5	4.0.0	5.0.0
12-2004	26				Version for Release 6	5.0.0	6.0.0
06-2007	36				Version for Release 7	6.0.0	7.0.0
12-2008	42				Version for Release 8	7.0.0	8.0.0
12-2009	46				Version for Release 9	8.0.0	9.0.0

History

Document history		
V9.0.0	January 2010	Publication