Technical Specification Group Services and System Aspects
Meeting #10, Bangkok, Thailand, 11-14 December 2000

*TSGS#10(00)0559*
Agenda Item: 7.4.1

# Presentation of Specification to TSG

| | |
|---|---|
| **Presentation to:** | **TSG SA Meeting #10** |
| **Document for presentation:** | **TS 26.191, Version 1.0.0** |
| **Presented for:** | **Information** |
| **Agenda Item:** | **7.4.1** |

**Title of document:**

**AMR Wideband Speech Codec; Error concealment of erroneous or lost frames**

# 3G TS 26.191 V1.0.0 (2000-12)

*Technical Specification*

**3rd Generation Partnership Project;**
**Technical Specification Group Services and System Aspects;**
**Mandatory Speech Codec speech processing functions AMR Wideband Speech Codec; Error concealment of erroneous or lost frames**
**(Release 4)**

Keywords

3GPP, Wideband codec

**3GPP**

Postal address

3GPP support office address

650 Route des Lucioles - Sophia Antipolis
Valbonne - FRANCE
Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Internet

http://www.3gpp.org

# Contents

# Foreword

This Technical Specification has been produced by the 3GPP.

The present document defines an error concealment procedure, also termed frame substitution and muting procedure, of the wideband telephony speech service employing the Adaptive Multi-Rate – Wideband (AMR-WB) speech coder within the 3GPP system.

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 this TS, 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 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 specification;

# 1 Scope

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

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

# 2 Normative references

This document incorporates, by dated and undated reference, provisions from other publications. These normative references are cited in 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 document only when incorporated in it by amendment or revision. For undated references, the latest edition of the publication referred to applies.


[1] 3G TS 26.202"AMR Wideband Speech Codec; Interface to RAN".

[2] 3G TS 26.190"AMR Wideband Speech Codec; Transcoding functions".

[3] 3G TS 26.193"AMR Wideband Speech Codec; Source Controlled Rate operation".

[4] 3G TS 26.201"AMR Wideband Speech Codec; Frame structure".


# 3 Definitions and abbreviations

## 3.1 Definitions

For the purposes of this document, the following definition applies:

**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 this document can be found in the references.

## 3.2 Abbreviations

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

| | |
|---|---|
| AMR-WB | Adaptive Multi Rate - WideBand |
| AN | Access Network |
| BFI | Bad Frame Indication from AN |
| BSI_netw | Bad Sub-block Indication obtained from AN interface CRC checks |
| prevBFI | Bad Frame Indication of previous frame |
| PDFI | Potentially Degraded Frame Indication |
| RX | Receive |
| SCR | Source  Controlled Rate (operation) |
| SID | Silence Descriptor frame (Background noise) |
| CRC | Cyclic Redundancy Check |
| ECU | Error Concealment Unit |
| BFH | Bad Frame Handling |
| medianN | N-point median operation |

# 4 General

The purpose of the error concealment procedure is to conceal the effect of lost AMR-WB 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, SID_BAD or LOST_FRAME. 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 solution provided in paragraph 6 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-WB 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. When the frame is not received, the network shall indicate RX_TYPE = RX_LOST_FRAME in which case the BFI flag is set as well. 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 Erroneous or lost speech frames

Normal decoding of erroneous/lost speech frames would result in very unpleasant noise effects. In order to improve the subjective quality, erroneous/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. Subclause 6 provide example solution.

## 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. Subclause 6 provides example solutions.

# 6 Example ECU/BFH Solution

## 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 right-shifted by one. 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;
    if(State > 6)
        State = 6;
else
    State = State >> 1;
```

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 6, the processing depends on the **BFI** flag.

The procedure can be described as follows:

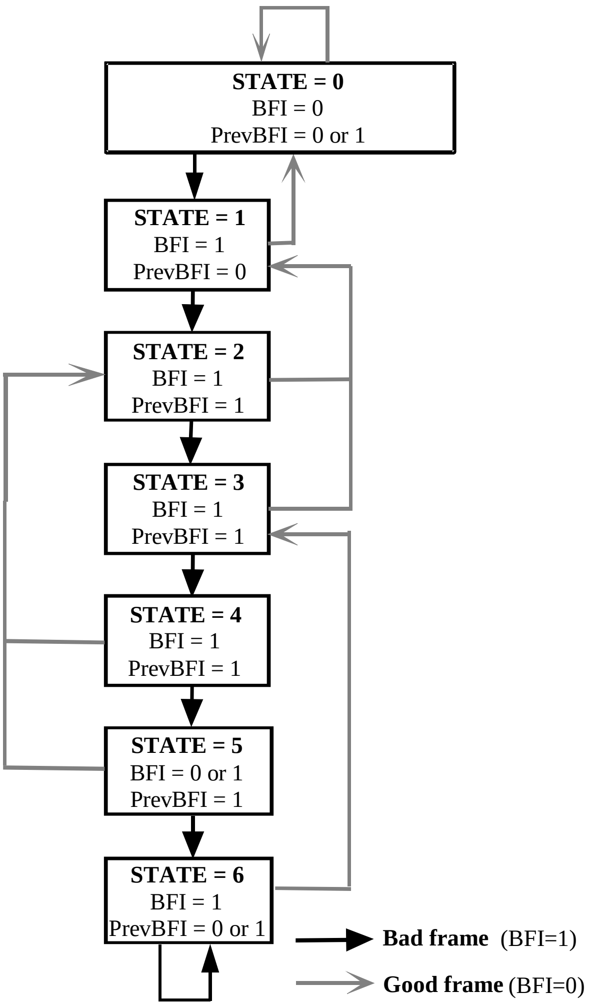


**Figure 1: State machine for controlling the bad frame substitution**

# 6.2 Substitution and muting of lost speech frames

## 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 normally 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 is used normally in the speech synthesis and fixed codebook gain are limited below the values used for the last received good subframe:

$$g^c(n) = \begin{cases} g^c_{received} & , g^c_{received} < 100 \; and \; g^c_{received} < g^c(n-1) \times 1.25 \\ g^c(n-1) & , otherwise \end{cases} \qquad (1)$$

where

$g^c_{received}$ = current decoded fixed codebook-gain

$g^c(n-1)$ = fixed codebook gain used for the last good subframe (BFI = 0)

$g^c(n)$ = fixed codebook gain to be used for the current frame.

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.

#### 6.2.3.1 LTP gain & fixed codebook gain concealment when RX_FRAMETYPE = SPEECH_BAD

The LTP gain and fixed codebook gain are replaced by attenuated values from the previous subframes:

$$g^p = P^p(state) * median5(g^p(n-1),...,g^p(n-5)) \qquad (2)$$

$$g^c = \begin{cases} P^c(state) * median5(g^c(n-1),...,g^c(n-5)) & ,\text{VAD\_HIST} \le 2 \\ median5(g^c(n-1),...,g^c(n-5)) & ,\text{VAD\_HIST} > 2 \end{cases} \qquad (3)$$

where:

$g^p$ = current decoded LTP gain,

$g^c$ = current decoded fixed codebook gain,

$g^p(n-1),...,g^p(n-5)$ = LTP gains used for the last 5 subframes,

$g^c(n-1),...,g^c(n-5)$ = fixed codebook gains used for the last 5 subframes,

*median5()* = 5-point median operation,

$P^p(state)$ = attenuation factor ($P^p(1)$ = 0.98, $P^p(2)$ = 0.96, $P^p(3)$ = 0.75, $P^p(4)$ = 0.23, $P^p(5)$ = 0.05, $P^p(6)$ = 0.01),

$P^c(state)$ = attenuation factor ($P^c(1)$ = 0.98, $P^c(2)$ = 0.98, $P^c(3)$ = 0.98, $P^c(4)$ = 0.98, $P^c(5)$ = 0.98, $P^c(6)$ = 0.70),

*state* = state number {0..6},

VAD_HIST is number of consecutive VAD=0 decisions.

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:

$$ener(0) = \frac{1}{4}\left[\sum_{i=1}^{4} ener(n-i)\right] - 3 \qquad (4)$$

#### 6.2.3.2 LTP gain & fixed codebook gain concealment when RX_FRAMETYPE = LOST_FRAME

The LTP gain and fixed codebook gain are replaced by attenuated values from the previous subframes:

$$g^p = P^p(state) * median5(g^p(n-1),...,g^p(n-5)) \qquad (5)$$

$$g^c = \begin{cases} P^c(state) * median5(g^c(n-1),..., g^c(n-5)) & \text{, VAD\_HIST} \le 2 \\ median5(g^c(n-1),..., g^c(n-5)) & \text{, VAD\_HIST} > 2 \end{cases} \quad (6)$$

where:

$g^p$ = current decoded LTP gain,

$g^c$ = current decoded fixed codebook gain,

$g^p(n-1),..., g^p(n-5)$ = LTP gains used for the last 5 subframes,

$g^c(n-1),..., g^c(n-5)$ = fixed codebook gains used for the last 5 subframes,

*median5()* = 5-point median operation,

$P^p(state)$ = attenuation factor ($P^p(1)$ = 0.95, $P^p(2)$ = 0.90, $P^p(3)$ = 0.75, $P^p(4)$ = 0.23, $P^p(5)$ = 0.05, $P^p(6)$ = 0.01),

$P^c(state)$ = attenuation factor ($P^c(1)$ = 0.50, $P^c(2)$ = 0.25, $P^c(3)$ = 0.25, $P^c(4)$ = 0.25, $P^c(5)$ = 0.15, $P^c(6)$ = 0.01),

*state* = state number {0..6},

VAD_HIST is number of consecutive VAD=0 decisions.

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:

$$ener(0) = \frac{1}{4}\left[\sum_{i=1}^{4} ener(n-i)\right] - 3 \quad (7)$$

### 6.2.3.3 ISF concealment

The past ISFs are shifted towards their partly adaptive mean:

$$ISF_q(i) = \alpha * past\_ISF_q(i) + (1-\alpha) * ISF_{mean}(i) \qquad i = 0..16 \quad (8)$$

where

$\alpha$ = 0.9,

$ISF_q(i)$ is ISF-vector for a current frame,

$past\_ISF_q(i)$ is ISF-vector from the previous frame,

$ISF_{mean}(i)$ vector is combination of adaptive mean and constant mean ISF-vectors in the following manner:

$$ISF_{mean}(i) = \beta * ISF_{const\_mean}(i) + (1-\beta) * ISF_{adaptive\_mean}(i), \qquad i = 0..16 \quad (9)$$

*where*

$\beta$ = 0.75,

$ISF_{adaptive\_mean}(i) = \frac{1}{3}\sum_{i=0}^{2} past\_ISF_q(i)$ and is updated whenever BFI =0.

$ISF_{const\_mean}(i)$ is a vector containing long time average of ISF-vectors.

### 6.2.3.4 LTP-lag concealment

The histories of five last good LTP-lags and LTP-gains are used for finding the best method to update.

### 6.2.3.4.1 LTP-lag concealment when RX_FRAMETYPE = SPEECH_BAD

The usability of the received LTP lag ( $Q_{lag}$ ) is defined as follows: (Predicts if the received lag is most probably very close to one that was sent and therefore its usage should not introduce any bad artifacts)

$$Q_{lag} = \begin{cases} 1 & ,T_{dif} < 10 \text{ and } \mathrm{T}_{\min} - 5 < T_{received} < \mathrm{T}_{\min} + 5 \\ 1 & ,\mathrm{g}^{\mathrm{p}}(n-1) > 0.5 \text{ and } \mathrm{g}^{\mathrm{p}}(n-2) > 0.5 \text{ and } \mathrm{T}\ (\mathrm{n}-1) - 10 < T_{received} < \mathrm{T}\ (\mathrm{n}-1) + 10 \\ 1 & ,g^{p}_{\min} < 0.4 \text{ and } g^{p}(n-1) = g^{p}_{\min} \text{ and } \mathrm{T}_{\min} < T_{received} < \mathrm{T}_{\max} \\ 1 & ,T_{dif} < 70 \text{ and } \mathrm{T}_{\min} < T_{received} < \mathrm{T}_{\max} \\ 1 & ,T_{mean} < T_{received} < \mathrm{T}_{\max} \\ 0 & ,otherwise \end{cases} \quad (10)$$

where:

$T(n-1)$ is LTP lag from the previous good frame,

$T_{dif} = \left| T_{received} - T(n-1) \right|$,

$T_{\min} = \min(T_{buffer})$,

$T_{\max} = \max(T_{buffer})$,

$T_{received}$ is received lag,

$g^{p}_{\min} = \min(g^{p}_{buffer})$,

$g^{p}$ is LTP gain of the current frame,

$g^{p}$ (-1) is LTP gain of the previous good frame,

$g^{p}$ (-2) is LTP gain of the frame before previous good frame,

$T_{mean} = average(T_{buffer})$

LPT lag value for the current frame is defined as follows:

$$T = \begin{cases} T_{received} & ,\mathrm{Q}_{\mathrm{lag}} = 1 \\ \frac{1}{3}\sum(T_{\max} + T_{\max-1} + T_{\max-2}) + RND(T_{\max} - T_{\max-2}) & ,\mathrm{Q}_{\mathrm{lag}} = 0 \end{cases} \quad (11)$$

where:

$T_{\max} = \max(T_{buffer})$,

$T_{\max-1}$ is second largest value in $T_{buffer}$ ,

$T_{\max-2}$ is second largest value in $T_{buffer}$ ,

$RND(x)$ is random value generated to range $\left[ -\frac{x}{2}, +\frac{x}{2} \right]$

### 6.2.3.4.2 LTP-lag concealment when RX_FRAMETYPE = LOST_FRAME

The usability of the LTP lag from last good frame ( $Q_{lag\_t-1}$ ) is defined as follows: (Predicts if the received lag is most probably very close to one that was sent and therefore its usage should not introduce any bad artifacts)

$$Q_{lag\_t-1} = \begin{cases} 1 & , g^{p}_{\min} > 0.5 \text{ and } \mathrm{T}_{\mathrm{dif}} < 10 \\ 1 & , \mathrm{g}^{\mathrm{p}}(n-1) > 0.5 \text{ and } \mathrm{g}^{\mathrm{p}}(n-2) > 0.5 \\ 0 & , otherwise \end{cases} \qquad (12)$$

where:

$g^{p}_{\min} = \min(g^{p}_{buffer})$,

$g^{p}$ (n-1) is LTP gain of the previous good frame,

$g^{p}$ (n-2) is LTP gain of the frame before previous good frame

LPT lag value for the current frame is defined as follows:

$$T = \begin{cases} T(n-1) & , \mathrm{Q}_{\mathrm{lag\_t\text{-}1}} = 1 \\ \frac{1}{3}\sum(T_{\max} + T_{\max-1} + T_{\max-2}) + RND(T_{\max} - T_{\max-2}) & , \mathrm{Q}_{\mathrm{lag\_t\text{-}1}} = 0 \end{cases} \qquad (13)$$

where:

$T(n-1)$ is LTP lag from the previous good frame,

$T_{\max} = \max(T_{buffer})$,

$T_{\max-1}$ is second largest value in $T_{buffer}$,

$T_{\max-2}$ is second largest value in $T_{buffer}$,

$RND(x)$ is random value generated to range $\left[-\frac{x}{2}, +\frac{x}{2}\right]$

### 6.2.4 Innovation sequence

When RX_FRAMETYPE = SPEECH_BAD, the received fixed codebook innovation pulses from the erroneous frame are used as they are received.

When RX_FRAMETYPE = LOST_FRAME, the received fixed codebook innovation pulses from the erroneous frame are not used and the fixed codebook innovation vector is filled with random signal (values limited to range [-1, +1]).

## 6.3 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

| Change history | | | | | | | |
|---|---|---|---|---|---|---|---|
| Date | TSG # | TSG Doc. | CR | Rev | Subject/Comment | Old | New |
| | | | | | | | |
| | | | | | | | |
| | | | | | | | |
| | | | | | | | |
| | | | | | | | |