

MAYAH Communications

Application Note 46

Error Concealment

1.	GENERAL DESCRIPTION	1
2.	TECHNICAL DESCRIPTION.....	1
3.	REQUIREMENTS	2
4.	ERROR CONCEALMENT METHODS.....	2
4.1	ZERO SUBSTITUTION (WITH MERGING).....	2
4.2	PACKET REPEAT (WITH MERGING)	2
4.3	PATTERN MATCHING.....	2
4.4	EXTRAPOLATION USING BURG'S PREDICTION FILTER LQ1	2
4.5	EXTRAPOLATION USING BURG'S PREDICTION FILTER LQ2	2
4.6	EXTRAPOLATION USING BURG'S PREDICTION FILTER LQ3	3
4.7	EXTRAPOLATION USING BURG'S PREDICTION FILTER MQ1	3
4.8	EXTRAPOLATION USING BURG'S PREDICTION FILTER MQ2	3
4.9	EXTRAPOLATION USING BURG'S PREDICTION FILTER HQ 1	3
4.10	EXTRAPOLATION USING BURG'S PREDICTION FILTER HQ 2	3
4.11	EXTRAPOLATION USING BURG'S PREDICTION FILTER SHQ 1	3
4.12	EXTRAPOLATION USING BURG'S PREDICTION FILTER SHQ 2	4
4.13	EXTRAPOLATION USING BURG'S PREDICTION FILTER SHQ 3	4
4.14	NO ERROR CONCEALMENT	4
5.	ERROR CONCEALMENT METHODS RECOMMENDATIONS.....	5
6.	INTERNAL INFORMATIONS.....	6

1. General description

Real-time audio data streamed over unreliable IP networks, such as the Internet, may encounter losses due to dropped packets or late arrivals. This paper lists the available options for error concealment schemes developed for streaming real-time audio data over the Internet.

If, despite all the measures and settings of bit rate, packet size and coding at the encoder side, there should be detected any transmission errors at the receiver/decoder side, they can be concealed by simple or very complex methods. Simple methods are for example muting with or without audio panel, i.e. muting of the flawless received audio signal in order to avoid so-called clicks.

Another simple method is to repeat the last correctly received packet by fading. Advanced algorithms are methods of prediction, that can achieve surprisingly good results according to given computing power.

2. Technical description

The error concealment method could be configured with the remote command "com_concealment". Please refer also to the Communications Reference Manual.

Due to different codec devices with different algorithms it could be necessary to configure a corresponding audio delay to handle the error concealment method; especially for low power and mobile devices.

Please note that the following concealment methods will be further developed.

3. Requirements

Firmware version 4.9.2.0 or higher must be running on the MAYAH codec!

4. Error concealment methods

4.1 Zero substitution (with merging)

Remote command: com_concealment 1

Substitute missing data by inserting zeros. Crossfading is used to reduce unwanted click sounds.

This method is available on all Mayah codecs which supports error concealment.

4.2 Packet repeat (with merging)

Remote command: com_concealment 2

Substitute missing data by inserting data from last received packet. Crossfading is used to reduce unwanted click sounds.

This method is available on all Mayah codecs which supports error concealment.

4.3 Pattern matching

Remote command: com_concealment 3

Substitute missing data with similar data from the past by using a pattern matching algorithm. Crossfading is used to reduce unwanted click sounds.

This method is only available on Mayah codecs which supports error concealment and have installed a 1600MHz CPU (or higher).

This method increase the audio delay by 70 milliseconds! ¹⁾

4.4 Extrapolation using Burg's prediction filter LQ1

Remote command: com_concealment 4

Substitute missing data by a warped prediction filter using past samples. This version should be used when minimal processing speed is required.

This method is only available on Mayah codecs which supports error concealment and have installed a 1600MHz CPU (or higher).

This method increase the audio delay by 6 milliseconds! ¹⁾

4.5 Extrapolation using Burg's prediction filter LQ2

Remote command: com_concealment 5

Substitute missing data by a warped prediction filter using past samples.

This method is only available on Mayah codecs which supports error concealment and have installed a 1600MHz CPU (or higher).

This method increase the audio delay by 14 milliseconds! ¹⁾

4.6 Extrapolation using Burg's prediction filter LQ3

Remote command: com_concealment 6

Substitute missing data by a warped prediction filter using past samples. This version should be used when a good reconstruction quality and fast processing speed is required.

This method is only available on Mayah codecs which supports error concealment and have installed a 1600MHz CPU (or higher).

This method increase the audio delay by 26 milliseconds! ¹⁾

4.7 Extrapolation using Burg's prediction filter MQ1

Remote command: com_concealment 7

Substitute missing data by a prediction filter using past samples. This version is optimized for stereo audio data where both channels have VERY similar content.

This method is only available on Mayah codecs which supports error concealment and have installed a 1600MHz CPU (or higher).

This method increase the audio delay by 70 milliseconds! ¹⁾

4.8 Extrapolation using Burg's prediction filter MQ2

Remote command: com_concealment 8

Substitute missing data by a prediction filter using past samples. This version is optimized for stereo audio data where both channels have VERY similar content.

This method is only available on Mayah codecs which supports error concealment and have installed a 1600MHz CPU (or higher).

This method increase the audio delay by 70 milliseconds! ¹⁾

4.9 Extrapolation using Burg's prediction filter HQ 1

Remote command: com_concealment 9

Substitute missing data by a prediction filter using past samples.

This method is only available on Mayah codecs which supports error concealment and have installed a 1600MHz CPU (or higher).

This method increase the audio delay by 110 milliseconds! ¹⁾

4.10 Extrapolation using Burg's prediction filter HQ 2

Remote command: com_concealment 10

Substitute missing data by a prediction filter using past samples.

This method is only available on Mayah codecs which supports error concealment and have installed a 1600MHz CPU (or higher).

This method increase the audio delay by 110 milliseconds! ¹⁾

4.11 Extrapolation using Burg's prediction filter SHQ 1

Remote command: com_concealment 11

Substitute missing data by a prediction filter using past samples.

This method is only available on Mayah codecs which supports error concealment and have installed a 1600MHz CPU (or higher).

4.12 Extrapolation using Burg's prediction filter SHQ 2

Remote command: com_concealment 12

Substitute missing data by a prediction filter using past samples.

This method is only available on Mayah codecs which supports error concealment and have installed a 1600MHz CPU (or higher).

4.13 Extrapolation using Burg's prediction filter SHQ 3

Remote command: com_concealment 13

Substitute missing data by a prediction filter using past samples.

This version should be the default to provide best overall reconstruction quality.

This method is only available on Mayah codecs which supports error concealment and have installed a 1600MHz CPU (or higher).

4.14 No error concealment

Remote command: com_concealment 0

Concealment deactivated.

¹⁾ The delay was measured on a 1600Mhz CPU! The real delay depends on the installed CPU and could be calculated by the $[\text{measured delay}] * 1600 / [\text{installed CPU in MHz}]$.

5. Error concealment methods recommendations

Sample rate	64 samples / packet	480 samples / packet	1024 samples / packet	2048 samples / packet
8000 Hz	4.10	4.10	4.10	4.10
16000 Hz	4.10	4.10	4.10 4.6	4.10 4.6
32000 Hz	4.6 4.5 4.4	4.10	4.10 4.6	4.10 4.6 4.3 4.2
44100 Hz	4.5 4.4	4.10 4.6	4.10	4.10 4.6 4.5 4.4
48000 Hz	4.5 4.4	4.10 4.6	4.10	4.10 4.6 4.5 4.4
96000 Hz		4.6 4.5 4.4	4.6 4.5 4.4 4.3	4.6 4.3 4.2

6. Internal informations

Parameter settings of all profiles are based on own simulation tests and the following general recommendations from different publications:

- AR models should only be estimated from "quasi-stationary" audio signals (20 - 50 ms)
- Rule of thumb: $N \geq 2 * p$ (even longer if warped methods are used)
- Satisfactory reconstruction results for standard Burg's method with $p = 1000$

According to above recommendations the sample buffer size N should always be:

$$N_{\min} \leq N \leq N_{\max}$$

$$N_{\min} = fs / (50 * 10^{-3} s)$$

$$N_{\max} = fs / (20 * 10^{-3} s)$$

According to above recommendations the model order should always be:

$$p \leq N_{\max} / 2$$

fs	Sampling frequency (in Hz)
lambda	Warping factor
p	Filter model order
N	Sample buffer size
G	Gap size