NOISE REDUCTION ON SPEECH CODEC PARAMETERS

Hervé Taddei, Christophe Beaugeant, Mickael de Meuleneire

Siemens AG, ICM MP, Haidenauplatz 1, 81675 Munich, Germany
Email: firstname.name@siemens.com

ABSTRACT

The transmission of speech in mobile or packet networks requires the use of a speech codec. In order to improve the quality of speech in a noisy environment, a noise reduction algorithm is used. This noise reduction can either be done as pre-processing before speech encoding or in the network by decoding the bitstream, performing the speech enhancement in the time and/or frequency domain and re-encoding the speech. Both methods are computationally expensive. In this paper a new approach to reduce environmental background noise by modifying the codec parameters is discussed.

1. INTRODUCTION

Where mobile phones are used, the background noise can impact the quality of the encoded speech (e.g. in crowded places, carkit applications). Speech codecs are not very robust against noise. Therefore, in mobile phones, noise reduction is usually done in the device before encoding the speech [1]. A few recent studies have focused on the interaction between noise reduction and speech coding [1, 2, 3] to enhance the global performance of the couple noise reduction/speech coding. However, these studies are limited to the interaction between two independent blocks. One further possibility is to integrate the noise reduction into the speech codec itself [4, 5]. Such embedded solutions can allow to noise reduction in the mobile phone itself, or alternatively in the network by doing noise reduction on the transmitted codec parameters [5].

In this article, we investigate such embedded systems. After introducing the AMR codec in section 2, we present our method in section 3. A CCR test has been conducted to evaluate the performance of this method compared to ”classical” based on short-term spectral weighting rules. The results are given in section 4.

2. OVERVIEW OF THE AMR CODEC

In mobile communications, air interface transmission requires a low bit rate and the end-user requires a high degree of intelligibility and quality of the transmitted speech. 3GPP chose the ACELP based AMR codec as the mandatory codec for coding of speech at $8kHz$ sampling frequency. This speech codec consists of a multi-rate speech coder, a source-controlled rate scheme including a Voice Activity Detection (VAD), a comfort noise generation system and an error concealment mechanism to compensate the effects of transmission errors and packet loss [6].

The AMR codec has a frame length of $20ms$. Each frame is divided into 4 subframes of equal length. The codec uses a 10^{th} order linear prediction filter. The Linear Prediction Coefficients (LPC), are computed for each frame by solving a linear system of equations. They are further quantized and transmitted as Line Spectral Pair (LSP). After filtering of the input signal by the LPC filter, a residual signal is obtained. This signal needs to be transmitted for reconstruction of the speech to the decoder. To do so, first an adaptive codebook search is performed on subframe basis leading to a pitch delay and an adaptive gain value. By subtracting the excitation of the adaptive codebook multiplied with its respective gain a new target signal is obtained. This target signal is used to process the fixed codebook search (fixed codebook index and fixed gain value). These parameters (pitch delay, fixed codebook index and both fixed and adaptive gains) are also transmitted to the decoder.

The decoder performs the synthesis of the speech using the transmitted parameters. The adaptive excitation is found by interpolating the past excitation multiplied by the adaptive gain. The fixed excitation is obtained by multiplying the codebook codevector by the fixed codebook gain. Both excitations are then summed up and enters the LPC synthesis filter. Finally, an optional post-processing algorithm is applied to enhance the quality of the reconstructed speech.

3. NOISE REDUCTION ALGORITHM

In this section, we consider a basic end-to-end transmission of speech through a network using the AMR codec at both ends. At the near-end side, the input signal $y(t)$ is assumed to be the sum of a speech signal $s(t)$ and of a noise signal $n(t)$. The encoder transmits a bitstream derived from the analysis of $y(t)$ every $20ms$ to the receiver. At the far-end side, the bitstream is decoded to reconstruct the speech.

0-7803-8484-9/04/$20.00 ©2004 IEEE

3.1. General Principle

In [4], some experiments have shown that the fixed gain parameter is linked to the noise amplitude. Indeed, the replacement of the fixed gain $y(t)$ by the gain factor of a less noisy signal $y_2(t) = s(t) + a \cdot n(t), a < 1$ leads to the subjective reduction of the noise. Such behaviour of the fixed gain is confirmed by considering the transfer function of the synthesis filter in the z-domain as in [5]. We can write that we have approximately:

$$H(z) = \frac{g_y(m)}{\left(1 - g_p(m) \cdot z^{-T(m)}\right)\left(1 + \sum_{i=1}^{M} a_i(m) \cdot z^{-i}\right)} \tag{1}$$

with M the length of the linear prediction filter ($=$ 10 in AMR), m the subframe index, a_i the LPC coefficients, g_p the adaptive gain, T the current pitch delay and g_y the fixed gain value.

With this formula, the fixed gain can be seen as a multiplicative factor applied to the signal. Accordingly, reducing g_y involves the reduction of the amplitude of the signal. As a result, by applying a weighting factor to the fixed gain we may expect to reduce the noise. This weighting factor has to be ruled by a noise or Signal to Noise Ratio (SNR) dependent law. Basically, for high SNR the weighting factor should tend to 1 and for low SNR to 0.

Most of the noise reduction methods are based on short-term spectral weighting rules [1, 2, 3], where weighting factors are applied to the amplitude of the noisy signal in the frequency domain. For each frame and for each frequency band, the amplitude of the signal is modified according to the noise level or the SNR at this frame. The interpretation of the fixed gain and the idea of applying a weighting rule to this gain lead us to make a parallel between the weighting rule applied in the frequency domain and the weighting rule that we may use in the parameter domain. Accordingly, we design our noise reduction as depicted in the fig. 1 where we used the classical approach of noise reduction solution in the frequency domain:

- Estimation of the fixed gain g_n that would be obtained through the coding of the signal $n(t)$.

- Applying the attenuation rule γ_c to the fixed gain of the noisy signal g_y. The estimated gain $\hat{g_s}$ is obtained through $\hat{g_s}(m) = \gamma_c(m) \cdot g_y(m)$. $g_y(m)$ is replaced in the bitstream by $\hat{g_s}(m)$.

- Controlling the amount of noise reduction and the artefacts introduced by applying a regulator post-filter.

In the following sub-sections, we describe the three different steps mentioned in this list.

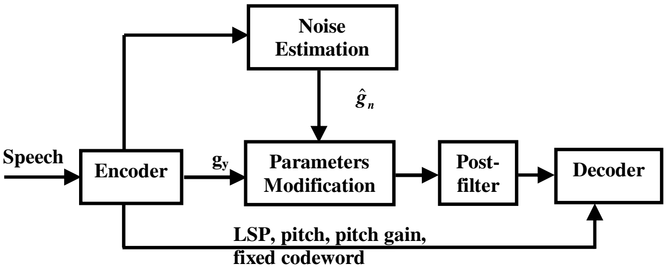

Fig. 1. Schematic of our proposed method

3.2. Minimum Statistic

To estimate g_n, we based our analysis on the transposition of the minimum statistic [7] from the frequency domain into the parameter domain. This method in the frequency domain assumes that the noise amplitude can be seen as a spectral floor of the signal. Our transposition assumes accordingly that the fixed gain g_n , interpreted as the amplitude of the noise, is the floor of the fixed gain of the noisy signal. As a result, finding the minimum of g_y leads to estimate g_n. The minimum of the fixed gain of the noisy signal is computed as follow:

- A smoothing factor is applied to g_y:

$$P(m) = \alpha(m).P(m-1) + (1 - \alpha(m))g_y^2(m) \tag{2}$$

- The minimum is searched within a window of D samples:

$$P_{min}(m) = min(P(m), ..., P(m-D)) \tag{3}$$

- The bias introduced by the determination of this minimum is compensated by an overestimation factor, $over$, so that the estimation of the fixed gain of the noise $\hat{g_n}$ becomes:

$$\hat{g_n}^2(m) = over.P_{min}(m) \tag{4}$$

In order to make $P(m)$ depending on the SNR, the smoothing factor $\alpha(m)$ in Eq. 2 is small during speech period and close to one during noise only periods. This is achieved according to the rule:

$$\alpha(m) = \max\left[\alpha_{min}, \frac{\alpha_{max}}{1 + \left(\frac{P(m-1)}{\hat{g_n}^2(m-1)} - 1\right)^2}\right] \tag{5}$$

Accordingly, if $P(m-1) >> \hat{g_n}^2(m-1)$, which can be interpreted as a period of high SNR at the frame $m-1$, the factor alpha tends to α_{min}, typically 0.3. This leads to slow up-date of $P(m)$. Conversely, if $P(m-1) \approx \hat{g}_n^2(m-1)$ (low SNR), the factor tends to α_{max}, typically 0.95.

3.3. Gain modification algorithm

In this section, the weighting factor γ_c applied to the fixed gain g_y is depicted. g_y can be seen as a function of the speech fixed gain g_s that would be computed if $s(t)$ were coded and of the noisy gain g_n:

$$g_y(m) = f(g_s(m), g_n(m)) \tag{6}$$

We have assumed first that f can be approximated by the linear additive function so that:

$$g_y(m) = g_s(m) + g_n(m) \tag{7}$$

Such an assumption is justified in noise only period and can be expected for high and low SNR. Indeed, for high (resp. low) SNR, the noisy signal can be roughly approximated by $y(t) = s(t)$ (resp. $n(t)$) so that the fixed gain g_y is about the speech (resp. noise) fixed gain. Nevertheless, such approximation of the function f means that our analysis is biased at least for 'average' SNR (i.e. when speech and noise have the same energy).

In order to reduce the bias influence, we consider the following relationship between g_y, g_s and g_n:

$$g_y^{\delta}(m) = g_s^{\delta}(m) + g_n^{\delta}(m) \tag{8}$$

This equation has the same properties as Eq. 7, but has the advantage that the approximation $g_y^{\delta} = g_s^{\delta}$ can be made for lower SNR if $\delta > 1$. Indeed, for a fixed value of g_n, we can consider that $g_s^{\delta} >> g_n^{\delta}$ for smaller values of g_s compared to values needed to assess $g_s >> g_n$. As a result, using Eq. 8 instead of Eq. 7 reduces the bias during high SNR periods, which leads in practice to better performance of our system during speech periods. Conversely, using the same argument for $\delta < 1$, the bias decreases in low SNR condition.

Accordingly, in order to avoid the important bias included in Eq. 7, we have based our approximation of the function between g_y, g_s and g_n on Eq. 8 with variable delta. In order to determine in which area of SNR we stand, the value of $\delta(m)$ is linked to the value of γ_c of the previous subframe:

$$\delta(m) = \begin{cases} \delta_1 & \text{if} \quad \gamma_c(m-1) \geq 0.5 \\ \delta_2 & \text{if} \quad \gamma_c(m-1) < 0.5. \end{cases} \tag{9}$$

with typically $\delta_1 = 2$ and $\delta_2 = 0.75$.

Resulting from the approximation in Eq. 8, an estimate of the gain g_s can be obtained through:

$$\hat{g}_s^{\delta(m)}(m) = g_y^{\delta(m)}(m) - g_n^{\delta(m)}(m) \tag{10}$$

Hence the weighting factor applied to g_y is:

$$\gamma_c(m) = SNR_{\delta}(m)/(1 + SNR_{\delta}(m)) \tag{11}$$

with $SNR_{\delta}(m) = g_s^{\delta(m)}(m)/g_n^{\delta(m)}(m)$ estimated by the following recursive formula:

$$\widehat{SNR}_{\delta}(m) = \beta\left(\frac{\hat{g}_s(m-1)}{\hat{g}_n(m)}\right)^{\delta(m)} + (1-\beta)\left(\frac{g_y(m)}{\hat{g}_n(m)}\right)^{\delta(m)} \tag{12}$$

3.4. Regulator post-filter

As already discussed above, the main drawback of our system is the bias involved by our approximation of the function f. Despite the bias compensation introduced by our variable definition of $\delta(n)$, speech signal processed by the weighting factor γ_c of Eq. 11 presents the drawback of reducing the energy of the signal during speech periods. A post-filter is thus introduced to keep the fixed gain of the processed signal $\hat{g}_s$ as near as possible to the fixed gain g_y in speech periods (high SNR).

For this purpose, we introduce the computation of the overall energy of the signal before and after processing (E_u and E'_u respectively) defined as:

$$E_u(m) = \sum_{i=1}^{N} \left(g_p(m) \cdot v_i(m) + \cdot g_y(m) \cdot c_i(m)\right)^2 \tag{13}$$

$$E'_u(m) = \sum_{i=1}^{N} \left(g_p(m) \cdot v_i(m) + \gamma_c(m) \cdot g_y(m) \cdot c_i(m)\right)^2 \tag{14}$$

where $v_i(m)$ and $c_i(m)$ stand for the adaptive codebook excitation and fixed codebook excitation respectively, N for the number of samples per subframe. Depending on the value of E_u and E'_u, the weighting factor is compensated so that the fixed gain stays unchanged for high SNR:

$$\gamma_c(m) = \begin{cases} \gamma_c(m) & \text{if} \quad 10\log_{10}\left(\frac{E_u}{E'_u}\right) \geq Th_{dB} \\ 1 & \text{if} \quad 10\log_{10}\left(\frac{E_u}{E'_u}\right) < Th_{dB} \end{cases} \tag{15}$$

with typically $Th_{dB} = 1$.

4. CCR TEST

In order to evaluate the quality of our noise reduction we compared it to a noise reduction based on short term spectral weighting rule [1]. For this purpose, we conducted a CCR test [8].

A set of original speech files were derived from eight different English speakers (four males, four females). These speech files were corrupted by additive noise (car noise and babble noise with an overall SNR of 10 dB and 20 dB). For these four conditions, the corrupted speech files were processed by algorithms A (the one depicted in this article) and B ([1]). For both algorithms, we used the mode 12.2 $kbit/s$

Condition	Mean score A/B
Babble noise at 10dB	-0.41
Car noise at 10dB	-0.23
Babble noise at 20dB	-0.12
Car noise at 20dB	-0.01
Global	-0.19

Table 1. Results of the CCR test.

of the AMR. Eight listeners took part in the test and had to rate the speech files with a score between -3 and +3.

Figure 2 depicts the histograms of the CCR score for each condition (babble noise 10 dB, babble noise 20 dB, car noise 10 dB, car noise 20 dB), as well as the associated interpolated Gaussian distributions according to mean and standard deviation obtained within this test. A positive value (a negative value respectively) corresponds to ”A is better than B” (”B is better than A” respectively). The results show a very small preference for algorithm B. This preference is all the less significant that the overall SNR is high. Moreover, the gap between A and B is lower for the car noise than for the babble noise. In any case, the results of the test show that both algorithms provide about the same quality on the processed signal. The absolute mean score values are indeed lower than 0.5. This conclusion is confirmed by our informal listening tests and by the comments made by listeners who took part in the CCR test. It was assessed that it was difficult to differentiate between the algorithms.

5. CONCLUSION

In this paper, we proposed a new noise reduction algorithm working on the speech codec parameters. A CCR test showed that this method provides already good results when compared to ”classical” methods based on spectral weigthing rule. The results of the CCR test show that both algorithms produced an enhanced signal with an equivalent listening quality. They show the feasibility of noise reduction system embedded in CELP speech codec. Our current method has the advantages of being of very low complexity and to be able to work in the network without presenting the transcoding disadvantage. The bitstream does not need to be decoded and reencoded, just the fixed gain parameter is modified, the other parameters stay the same. Nevertheless, compared to spectral methods, the resolution of our technique is low as only one parameter, the fixed gain, is modified per subframe (each $5ms$). Accordingly, further investigations are needed in order to base the noise reduction system not only on fixed gain modification but on other parameters, such as the LPC coefficients. This would further enhance the quality of our proposed method.

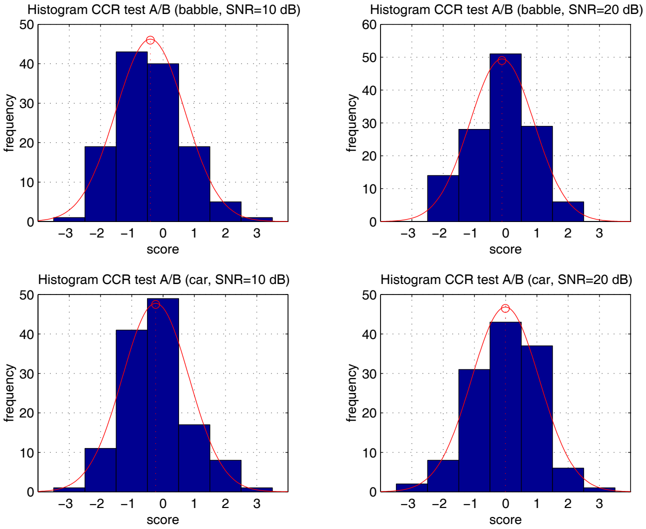

Fig. 2. Results of the CCR test.

6. REFERENCES

[1] P. Jax, R. Martin, P. Vary, M. Adrat, I. Varga, W. Frank, and M. Ihle, “A noise suppression system for the AMR speech codec,” in *Proc. KONVENS*, 2000.

[2] R. Martin and R.V. Cox, “New speech enhancement techniques for low bit rate speech coding,” in *Proc. IEEE Workshop Speech Coding*, 1999, pp. 165–167.

[3] D. Virette, P. Scalart, and C. Lamblin, “Analysis of background noise reduction techniques for robust speech coding,” in *EUSIPCO*, 2002.

[4] N. Duetsch, H. Taddei, C. Beaugeant, and T. Fingscheidt, “Noise reduction on speech codec parameters,” in *5th ITG International Conference on Source and Channel Coding*, 2004.

[5] Ravi Chandran and Daniel J. Marchok, “Compressed domain noise reduction and echo seppression for network speech enhancement,” in *Proc.of the 43rd IEEE Midwest Symposium on Circuits and Systems*, August 2000, vol. 1, pp. 10–13.

[6] 3GPP TS 26.071, *Mandatory Speech Codec speech processing functions; AMR speech codec; General Description*, June 2002.

[7] R. Martin, “Spectral subtraction based on minimum statistics,” in *EUSIPCO*, 1994, pp. 1182–1185.

[8] ITU-T, *Methods for Subjective Determination of Transmission Quality*, Aug. 1996.