A ROBUST BACKWARD ADAPTIVE QUANTIZER *

Dominique MARTINEZ[(1,2)] & **Woodward YANG**[(2)]

(1) Laboratoire d'Analyse et d'Architecture des Systèmes
LAAS - CNRS, 7 Av. du Col. Roche, 31077 Toulouse, France.

(2) Harvard University
Division of Applied Sciences
29 Oxford St., Cambridge, MA 02138, USA.

Abstract

This paper describes an adaptive encoder/decoder for efficient quantization of nonstationary signals. The system uses a robust backward adaptive encoding method such that the adaptation of the encoder and decoder is only determined by the transmitted codeword and does not require any additional side information. By incorporating a forgetting parameter, the quantizer is also robust to transmission errors and encoder/decoder mismatches. It is envisioned that practical applications of this algorithm can be used in the design of adaptive codecs (A/D and D/A converters) or as an efficient source coding algorithm for transmission of digitized speech.

1 INTRODUCTION

A typical waveform coder transmits information across a communication channel such that the resulting reconstruction of the signal at the decoder, say u', is as close as possible to the original source u. In this communication context, a quantizer $Q(.)$ can be viewed as an encoder/decoder pair connected to a digital channel. The encoder $\mathcal{E}$ converts a sampled version u_n of the source into a digital form $I_n = \mathcal{E}(u_n)$, sends it over the channel $\mathcal{C}$ to the decoder $\mathcal{D}$ which reconstructs the signal $u'_n = \mathcal{D}(I'_n)$ according to the codeword $I'_n = \mathcal{C}(I_n)$ received. Note that $\mathcal{C}(.)$ corresponds to the deterministic identity mapping for a noiseless channel such that $I'_n = I_n$ and to a probabilistic mapping for a

* To appear at the IEEE Workshop on Neural Networks for Signal Processing (NNSP'95).

noisy channel. Then, as shown in Fig. 1, $u'_n = Q(u_n) = \mathcal{E} \text{ o } \mathcal{C} \text{ o } \mathcal{D}(u_n)$. Throughout the paper, signals or parameters of the decoder which are similar to those of the encoder are denoted with primes.

A quantizer is optimal with respect to the probability density function (p.d.f) $p(u)$ of the source to be encoded if it minimizes a r-th power distortion measure $d(u, u') = D_r \equiv \int \mid u - u' \mid^r p(x) \quad dx$ with $0 < r \leq \infty$. In quantization, the most commonly used powers are $r = 1$ (mean absolute error) (see *e.g.* [1]), $r = 2$ (mean squared error) (see *e.g.* [2, 3, 4]) and $r = \infty$ (mean maximum absolute error). In case $p(x)$ is not known, the most widely used design algorithm for scalar quantizers is the (standard) Lloyd I algorithm, which can be extended for minimizing r-th power law distortion (generalized Lloyd I) [5]. However, because they are batch algorithms, the design of the quantizer can only begin after the entire training set is available. By consequence, these algorithms are not able to accommodate on-line changes in the input p.d.f.

In many applications, communication systems may have to carry signals of changing statistics, e.g. speech inputs with different variances. The most efficient way to handle nonstationary inputs is to continuously adapt the encoder/decoder pair in such a way as to match observed local statistics of the input sequence. For performing adaptive quantization, a number of researchers have developed unsupervised competitive learning algorithms (for references, see [6]), of which the Kohonen algorithm is one of the most well-known. Because the weight updates are a function of the input u, this scheme is referred to as forward adaptation and suffers the serious drawback that an excessive amount of side information is required to transmit the updates to the decoder [5].

On the contrary, backward adaptation is primarily of interest for designing adaptive quantizers because it does not require transmitting additional bits. As depicted in Fig. 1, the digital codewords I' and I are now used instead of the input u to adjust the quantizing parameters. Commonly used backward adaptive algorithms are gain or step size adaptive and expand or contract the dynamic range of a time invariant quantizer according to an estimate of the signal variance [7]. These algorithms assume symmetrical zero-mean p.d.f.s of known shape and therefore are not ideal devices for quantizing non stationary inputs.

A new backward adaptive algorithm has been introduced for buiding scalar quantizers "on-line" and without assuming any particular p.d.f's shape [8]. The learning rule, called generalized Boundary Adaptation Rule (BAR_r), minimizes r-th power law distortion D_r in the case of high resolution quantization. Unfortunately, in a communication situation, such a backward adaptive process is sensitive to transmission errors. If $I'_n \neq I_n$ is received at time n, then the decoder will not adapt appropriately and the resulting mismatch between encoder and decoder will remain in the system unless an alignment or recalibration is performed.

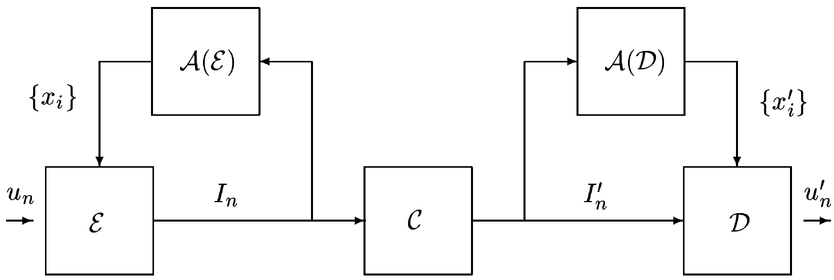

Figure 1: Block diagram of a backward adaptive quantizer. Encoder ($\mathcal{E}$), Channel ($\mathcal{C}$), Decoder ($\mathcal{D}$), Encoder and Decoder Adaptation ($\mathcal{A}(\mathcal{E})$ and $\mathcal{A}(\mathcal{D})$). Prime notations stand for the decoder's parameters. The decoder presents a symmetrical structure in order to track the encoder adaptive parameters.

In this article, we introduce a modified Boundary Adaptation Rule which is robust to channel errors and initial encoder/decoder mismatches.

2 QUANTIZER DESCRIPTION AND ANALYSIS

A regular N-point scalar quantizer $Q(.)$ maps the scalar-valued input u_n into one of N reconstruction levels. The quantizing system of interest is illustrated in Fig. 1. The encoder is specified by an ordered set of boundary points $x_0 < \cdots < x_{i-1} < x_i < \cdots < x_N$ delimiting N disjoint quantization intervals $R_1, \cdots, R_i, \cdots, R_N$, with $R_i = [x_{i-1}, x_i)$. Its output I_n is usually defined by the N-bit binary vector $(\mathbb{1}_{R_1}, \cdots, \mathbb{1}_{R_N})$ in which $\mathbb{1}_{R_i}(u_n) = 1$ if $u_n \in R_i$ and $= 0$ otherwise. In a communication situation however it is usually mapped into a more compact R-bit codeword representation with $R \geq \log_2 N$ for transmission. The decoder has a structure similar to the encoder but with prime notations standing as its parameters. The decoding operation corresponds to $u' = \sum_{i=1}^{N} \mathbb{1}'_{R_i} y'_i$. In high resolution quantizers, the number of quantization intervals N is very large so that the interval lengths are very small and the reconstruction levels y'_i can be approximated by the midpoints of their corresponding quantization intervals: $y'_i = (x'_{i-1} - x'_i)/2$.

The boundary points delimiting the quantization intervals are therefore the only parameters to adapt. At the encoder, the backward adaptation of x_i can be written as:

$$\Delta x_i(n) = x_i(n+1) - x_i(n) = \eta f(I_n) \tag{1}$$

where η is the learning rate, a positive scalar. In its simplest form, the generalized unsupervised learning rule called BAR_r (generalized Boundary Adaptation Rule) reduces to Eq. (1) with:

$$f(I_n) = \delta_{i+1}^r \mathbb{1}_{R_{i+1}} - \delta_i^r \mathbb{1}_{R_i} \tag{2}$$

where $\delta_i = x_{i-1} - x_i$ is the length of the interval R_i. Assuming that for input u_n, $1\!\!1_{R_i} = 1$, then only the interval R_i is reduced in size by increasing x_{i-1} and decreasing x_i. A faster rule, called $FBAR_r$, is obtained by updating all boundary points each time an input is presented:

$$f(I_n) = \sum_{k=i+1}^{N} \delta_k^r \frac{1\!\!1_{R_k}}{N-i} - \sum_{k=1}^{i} \delta_k^r \frac{1\!\!1_{R_k}}{i} \qquad (3)$$

$FBAR_r$ and BAR_r minimize r-th power law distortion [8]. At convergence, all the N quantization intervals R_i will have the same distortion $D_r(i) = D_r/N$. This property guarantees an optimal high resolution quantization [9].

To track the encoder parameters, the same learning equations are used at the encoder but, again with prime notations.

3 A ROBUST BOUNDARY ADAPTATION RULE

In the case of a noisy channel, I'_n may be different to I_n resulting in a mistracking of the encoder. Indeed, with Eq. (1), the mismatch from inappropriate adaptations will remain within the system. To help the system readjust, a leakage or "forgetting" factor β needs to be introduced in the learning rule. Eq. (1) therefore becomes:

$$\Delta x_i(n) = \eta f(I_n) - \beta x_i(n). \qquad (4)$$

To assess the effect of β on the robustness of the system, lets $d_n(.)$ denote the difference at time n between two similar parameters, one at the encoder and the other at the decoder. $d_n(x_i) = x_i(n) - x'_i(n)$ can be calculated recursively and expressed as:

$$d_n(x_i) = (1-\beta)^n d_0(x_i) + \eta \sum_{k=0}^{n-1} (1-\beta)^{(n-k-1)} d_k(f(I))$$

Clearly, in the absence of transmission errors, $d_n(x_i) = (1-\beta)^n d_0(x_i) \to 0$ when $\beta < 1$ and $n \to \infty$ and the initial encoder/decoder difference $d_0(x_i)$ decays with time. Similarly, the offset $d_k(f(I))$ from each transmission error $I'_k \neq I_k$ decays exponentially with time.

Eq. (4) is equivalent to Eq. (1) at convergence only if $\beta << \eta$. However a high value of β is necessary to improve performance in the presence of transmission errors. Furthermore, larger values of β in Eq. (4) also modify the basic performance criterion which will be no longer an r-th power law distortion D_r.

To maintain the ability of fully recovering from transmission errors while minimizing D_r, we propose the following modified learning rule

$$\Delta x_i(n) = \eta f(I_n) - \beta(x_i(n) - x_i{}^\star) \tag{5}$$

in which x_i now decays to some (fixed a-priori) nominal value $x_i{}^\star$. Note that Eq. (5) is identical to (4) when $x_i{}^\star = 0$ and to (1) when $\beta = 0$. More interesting is the fact that, at convergence, Eq. (5) and (1) are equivalent if $x_i{}^\star = E[x_i]$ as found by (5) at convergence, no matter the value of β.

The nominal boundary point values should be chosen equivalent for the decoder and for the encoder and be comparable to those given by a quantizer optimized for the long term p.d.f. at hand (e.g. a μ-law quantizer for speech coding or a quantizer optimized for a gaussian p.d.f in case of differential waveform coding assuming gaussian residuals). For high values of β, forgetting will be predominant over adaptation and the boundary points will eventually fluctuate around the nominal values specified by the user.

4 APPLICATION TO WAVEFORM CODING

We have previously shown that, for large N, $FBAR_r$, Eq. (1) and (3), outperforms the generalized Lloyd I algorithm in minimizing r-th power law distortion D_r with $r = 1$ and 2, for a gaussian and an exponential p.d.f. [8]. Here, we show the performance of the modified algorithm (Eq. (5)) in quantizing speech signals with noisy channels. $FBAR_r$ is preferred above BAR_r in waveform coding applications because convergence is $\mathcal{O}(N)$ time faster as noted in [8], and for this reason Eq. (3) will be used in conjunction with Eq. (5) in numerical simulations.

The performance of a quantizer is often assessed in terms of signal-to-noise ratio (SNR) specified in units of decibels (dB):

$$SNR = 10 \ \log_{10} \frac{E[u^2]}{D_2}$$

Another measure of interest reflecting temporal variations of SNR in waveform coding is the segmental signal-to-noise ratio (SEGSNR) which is simply a time-average of the ratio computed for each segment of the input sequence. Typically, appropriate segment length in speech coding is in the order of 16ms [7]. Although the maximization of both SNR and SEGSNR implies the minimization of the mean square error D_2, $FBAR_r$ with $r = 1$ has been employed in our adaptive quantizer for simplicity. To avoid dependency on a particular choice of fixed overload points, x_0 and x_N were taken after each time step as follows: $x_0 = x_1 - \delta_2$ and $x_N = x_{N-1} + \delta_{N-1}$.

Ideally, in presence of a locally stationary input like a speech signal, an adaptive quantizer would adapt quickly to abrupt changes in the input p.d.f. and stop adapting in the presence of a stationary segment. Fig. 2 shows the effect of the value of the learning rate η on the speed of adaptation of a typical speech signal originated from TIMIT database.

A value of $\eta = 0.49$ causes intantaneous adaptation, i.e. sudden changes following the local waveform peaks. On the contrary, a value of $\eta = 0.049$ causes syllabic adaption, i.e. slower changes that do not follow the local peaks, but only their envelope. The value of η mediates therefore the tradeoff between instantaneous and syllabic adaptation. We have found that a value of $\eta = 0.2$ is acceptable such that $FBAR_1$ tracks the changing statistics of speech with appropriate speed and with high values of SEGSNR. Table 1 compares the performance of $FBAR_1$ ($\eta = 0.2, \beta = 0.0$) with those of fixed quantizers such as uniform and μ-law. The results indicate that $FBAR_1$ gives at least a 4 dB gain in SNR over μ-law. In addition, unlike non-adaptive quantizers, $FBAR_1$ maintains a fairly constant and non-negative SNR-versus-time characteristic for all input speech segments. The resulting SEGSNR value attained is therefore superior to that of a μ-law or uniform quantizer, especially at low rates: 12.71 dB ± 0.044 instead of 5.11 and -5.46 dB for 3 bits, respectively.

Mismatches and long term drifts in hardware characteristics are a common drawback in analog-to digital converters. The self-correcting capability of our adaptive quantizer from initial encoder/decoder mismatches is illustrated in Table 2 for an initial mismatch $m_0 = 1.5$ and for different values of the forgetting factor β. The quantizer mismatch is defined at time n by $m_n = \sum_{i=1}^{N} [d_n(x_i)]^2$. In this particular example, the time average mismatch M was approximately $(m_0 10^{-6})/\beta$ for the range $10^{-5} \le \beta \le 10^{-1}$. For high values of β, the SNR and SEGSNR values are very closed to those obtained with a fixed quantizer having nominal values as its boundary points. This is easily verified by comparing the SNR values given in Table 2 for $\beta = 0.1$ and μ-law as nominal values, to those given in Table 1 for a fixed 4-bit μ-law quantizer. Similarly for $\beta = 0.1$ and zero as nominal values, the algorithm yields a constant zero output signal $u' = 0$ such that $D_2 = E[(u - u')^2] = E[u^2]$ and therefore SNR=0 dB. Use of a forgetting factor, together with μ-law nominal values, in $FBAR$ increases the coder robustness and performance: SNR=16.79 dB and SEGSNR=14.09 dB for $\beta = 10^{-3}$ over SNR=1.20 dB and SEGSNR = -6.76 dB for $\beta = 0$.

Quantizer type ↓	$R = \log_2 N$ bits →	3	4	5
Uniform	SNR SEGSNR	1.95 -5.46$^{(*)}$	8.64 1.21	15.06 7.74
μ-law ($\mu = 255$)	SNR $^{(**)}$ SEGSNR	8.44 5.11	13.51 11.52	19.67 18.10
$FBAR_1{}^{(***)}$	SNR SEGSNR	12.10 (± 0.052) 12.71 (± 0.046)	18.41 (± 0.057) 17.53 (± 0.044)	23.14 (± 0.091) 22.06 (± 0.062)

Table 1: SNR and SEGSNR values attained in 3-,4- and 5-bit coding of example speech signal. All entries in dB.
(*) The presence of zero amplitude input segments may yield occasionally extremely large negative SEGSNR values.
(**) Experimental SNR values for μ-law agree with those found by the theoretical formula $SNR_{\mu-\text{law}}(dB) = 6.02R - 10.1$ derived by Smith for $\mu = 255$ [10].
(***) For $FBAR_1$ results, the values given are averages over 20 runs with their standard deviations.

β	Nominal values $\{x_i^\star\}$	Average mismatch M	SNR (dB)	SEGSNR (dB)
0.0	——	0.15	1.20	-6.76
1E-1	0.0	1.2E-5	0.0	-0.23
1E-1	μ-law	1.2E-5	14.36	11.82
1E-2	μ-law	1.4E-4	16.11	13.09
1E-3	μ-law	1.5E-3	16.79	14.09
1E-4	μ-law	1.5E-2	10.66	9.40
1E-5	μ-law	0.94E-1	3.18	-4.61

Table 2: SNR and SEGSNR (dB) values attained in 4-bit coding of example speech signal in presence of an initial encoder/decoder mismatch $m_0 = 0.15$.

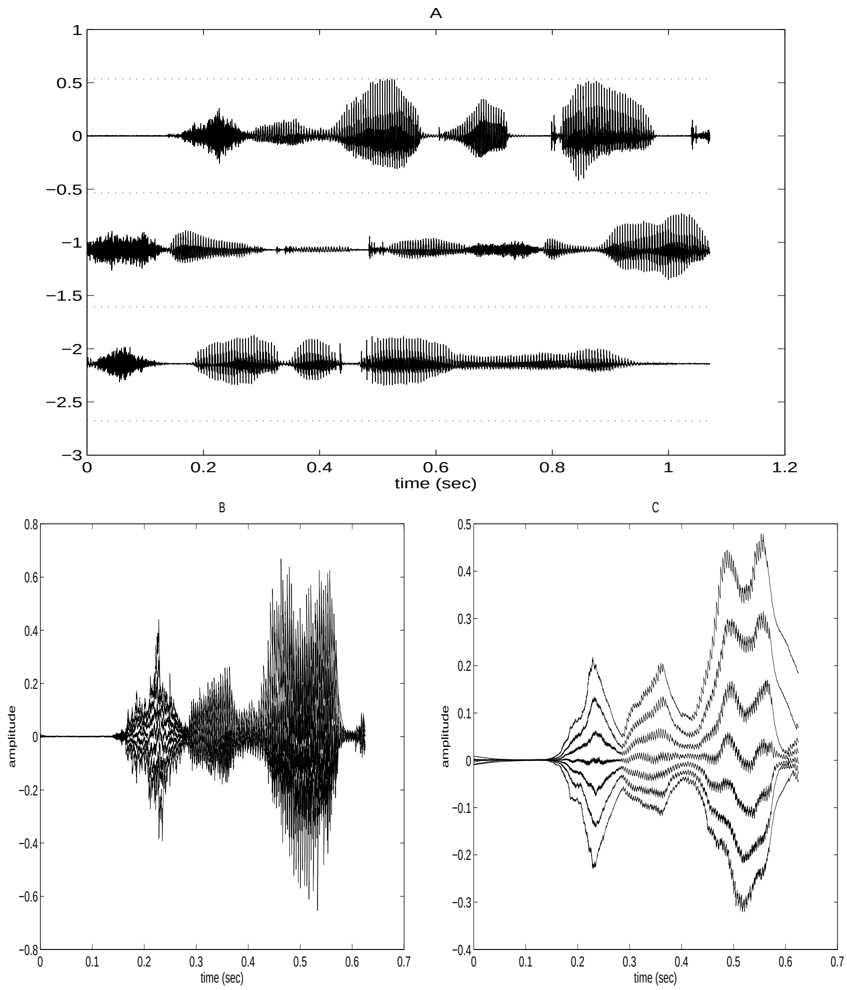

Figure 2: Instantaneous and Syllabic adaptation in quantizing speech signal.

Top: input speech sentence /She had your dark suit in greasy wash water all year/ of 51406 samples used in the simulations. *Bottom left:* Time evolution of the boundary points for a 3-bit instantaneous adaptive quantizer ($\eta = 0.49, \beta = 0.0, r = 1$). The quantizer has been adapted over the entire duration of the signal but only the first 0.6 seconds of the evolution are represented here. *Bottom right:* Time evolution of the boundary points for a 3-bit syllabic adaptive quantizer ($\eta = 0.049, \beta = 0.0, r = 1$).

The effect of transmission errors is evaluated by simulating a binary symmetric channel which produces errors with a probability $p_e = 10^{-3}$. Because the probability of one or more errors in the received R-bit codeword is approximately equal to Rp_e for small p_e [7], there were approximately 200 transmission errors over the entire duration of the signal. For the channel studied, performance results are given in Table 3 for a 4-bit quantizer with μ-law nominal values and different values of the forgetting factor β. Results clearly indicate that the use of a forgetting factor yields significant improvements in SNR and SEGSNR. It is reasonable to think that large displacements caused by the most significant bit (MSB) in error, most significantly contribute to a decrease in SNR. However, Table 3 reports that when the MSB is protected, improvements in adaptating are not greater than 1.5 dB when $\beta \neq 0$.

β	Average mismatch	Performance (dB)	
		SNR	SEGSNR
0.0	1.52 ± 1.68 [0.75 ± 1.13]	-6.66 ± 5.35 [-1.44 ± 6.67]	-7.25 ± 6.26 [-4.10 ± 6.51]
1E-1	1.30E-5 ± 2.0E-6 [0.80E-5 ± 1.0E-6]	12.63 ± 0.20 [13.28 ± 0.19]	9.13 ± 0.36 [10.38 ± 0.25]
1E-2	1.04E-4 ± 1.9E-5 [6.40E-5 ± 1.0E-5]	13.91 ± 0.32 [14.98 ± 0.15]	10.24 ± 0.30 [11.70 ± 0.18]
1E-3	1.12E-3 ± 4.8E-4 [8.05E-4 ± 3.2E-4]	15.35 ± 0.50 [16.70 ± 0.37]	11.36 ± 0.45 [12.92 ± 0.39]
1E-4	6.54E-2 ± 5.2E-2 [6.91E-2 ± 5.3E-2]	5.88 ± 3.43 [5.62 ± 3.12]	3.29 ± 3.25 [2.67 ± 2.36]
1E-5	5.07E-1 ± 4.4E-1 [6.11E-1 ± 6.9E-1]	-2.51 ± 4.15 [-2.61 ± 4.58]	-4.27 ± 4.69 [-4.58 ± 4.71]

Table 3: SNR and SEGSNR (dB) values attained in a 4-bit coding of example speech signal in presence of a noisy binary symetric channel with $p_e = 10^-3$. The quantizers have μ-law nominal values but different values of forgetting factor β. The values given are averages over 20 runs with their standard deviations. The values in brackets are given when the MSB is protected.

5 CONCLUSION

In this article, a related robust backward adaptive quantizer has been introduced for waveform coding in the presence of noisy channels. It has been shown that the forgetting term used in the learning rule dissipates transmission errors or initial quantizer mismatches while maintaining the ability to adapt itself to changes in the input p.d.f. and minimizing r-th power law distortion measure. Furthermore, if a hardware implementation is envisioned, the unsupervised learning rule will be able to adjust itself to initial mismatches and to long term drifts in component characteristics, hence avoiding the problem of pair-wise tuning parameters between encoders and decoders.

REFERENCES

[1] S.A. Kassam, "Quantization based on the mean-absolute-error criterion," **IEEE Trans. on Communications**, Vol. COM-26, pp. 267-270, 1978.

[2] S.P. Lloyd, "Least squares quantization in PCM's," **Bell Telephone Laboratories Paper**, Murray Hill, NJ, 1957.

[3] S.P. Lloyd, "Least squares quantization in PCM," **IEEE Trans. Inform. Theory**, Vol. IT-28, pp. 127-135, 1982.

[4] J. Max, "Quantizing for minimum distortion," **IRE Trans. on Inform. Theory**, Vol. IT-6, pp. 7-12, 1960.

[5] A. Gersho and R.M. Gray, **Vector quantization and signal compression**, Boston, Dordrecht, London: Kluwer, 1992.

[6] J. Hertz, A. Krogh, and R.G. Palmer, **Introduction to the theory of neural computation**, Reading MA: Addison-Wesley, 1991.

[7] N.S. Jayant and P. Noll, **Digital coding of waveforms**, Englewood Cliffs, New Jersey: Prentice-Hall, 1984.

[8] D. Martinez and M.M. Van Hulle, "Generalized Boundary Adaptation Rule for minimizing r-th power law distortion in the high resolution case," **Neural Networks**, Accepted for publication, 1995.

[9] A. Gersho, "Asymptotically optimal block quantization," **IEEE Trans. Inform. Theory**, Vol. IT-25, no. 4, pp. 373-380, July 1979.

[10] B. Smith, "Instantaneous companding of quantized signals," **Bell System Tech. J.**, Vol. 36, no. 3, pp. 653-709, 1957.