2013 Australian Communications Theory Workshop (AusCTW)

Ultra-low delay lossy audio coding using DPCM and block companded quantization

Gediminas Simkus, Martin Holters, and Udo Zölzer
Helmut Schmidt University — University of the Federal Armed Forces
Department of Signal Processing and Communications
Holstenhofweg 85, Hamburg, Germany
{gediminas.simkus,martin.holters,udo.zoelzer}@hsu-hh.de

***Abstract*—Real-time audio transmission requires good quality for a restricted channel capacity and minimum latency. An ultra-low delay audio coding scheme based on differential pulse code modulation (DPCM) and block companded quantization is presented. The prediction filter of the base backward DPCM codec is attained as a FIR filter in lattice structure. The proposed block based audio coding introduces an algorithmic latency below one millisecond and an overhead of less than a half bit per sample. The advantage of the block companded quantization with DPCM is the capability to follow rapid changes in the signal. Therefore, it significantly improves the perceptual audio quality compared to a plain DPCM coding scheme with an adaptive quantizer.**

I. Introduction

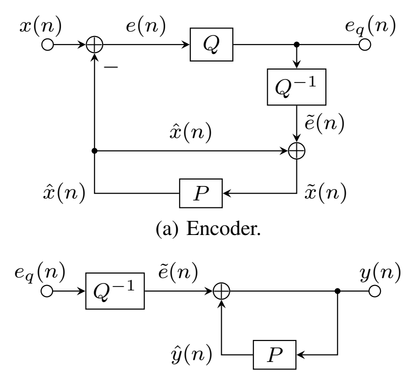
The state-of-the-art lossy audio coding schemes are able to achieve transparent audio quality at low bit-rates. Well known contemporary audio codecs like MPEG-Layer 3 (MP3) and MPEG-4 (AAC) provide excellent audio quality and are widely used in applications for the storage and broadcast of an audio signal. Audio codecs like MP3 or AAC due to subband coding and therefore block-wise processing lead to a large latency which can exceed 100 ms [1]. Large algorithmic delay makes a bidirectional communication for a real-time application, like wireless microphone and in-ear monitoring in live performance, in general not feasible. In [2] it is suggested that the requirement for algorithmic latency should be less than 5 ms.

Linear predictive coding (LPC) techniques are widely established in speech codecs [3], [4] to remove the redundancy from audio signals. Speech coding schemes introduce low to zero delay for a given bit rate, but do not yield sufficient audio quality for other audio signals. Significant audio quality improvements have been attained in [5] by extending the adaptive differential pulse code modulation (ADPCM), the principal part of the speech codec, by pre- and post-filtering.

II. The proposed system

The audio coding scheme proposed in this paper uses differential coding for the decorrelation of the input signal and can be seen as a replacement of the adaptive quantizer by block companded quantization [6] in the well-known base ADPCM coding scheme [7], [8]. The proposed system can be seen also as the extension of DPCM by the block compander. The prediction in the proposed coding system operates as backward-adapted DPCM, thus there is no need to transmit prediction coefficients from encoder to decoder. As the block compander is used, the decorrelated signal needs to be divided in to short blocks and normalized. A single block consists of less than twenty samples and therefore introduces negligibly short algorithmic delay to the coding scheme. The blocks are normalized and the normalization coefficient has to be transmitted to the decoder as side information. This overhead is small compared to the code word length per sample. Besides the mentioned drawbacks of the block companded quantization, the advantage compared to sample-by-sample adaptive quantization is that no clipping of the decorrelated signal appears. This allows for the proposed system to improve the perceptual audio quality by coding transient signals and provide audio quality near to those coding schemes which use noise shaping techniques [5], [9].

A. DPCM



(a) Encoder.

(b) Decoder.

Fig. 1. Structure of the DPCM Codec.

The base structure of the standard DPCM coding scheme is depicted in Fig. 1. This scheme corresponds to the codecs from [5], [7], [8]. Encoder and decoder contain two main blocks, the prediction filter and the quantizer. The prediction filter operates in backward-fed manner. First, for given current input $x(n)$ to the encoder, a current prediction $\hat{x}(n)$ is subtracted. The resulting residual signal is called prediction error $e(n) = x(n) - \hat{x}(n)$. The reconstructed prediction error $\tilde{e}(n) = Q^{-1}\big(e_q(n)\big)$ is added up with the current prediction $\hat{x}(n)$ in the encoder and $\hat{y}(n)$ in the decoder for the reconstruction of the current input signal, where $\mathrm{Q}^{-1}(\cdot)$ denotes the dequantization operation. This reconstructed signal $\tilde{x}(n) = \hat{x}(n) + \tilde{e}(n)$ and $y(n)$ is fed back to the encoder and decoder prediction filter, respectively. The prediction filter P uses $\tilde{x}(n)$ and $y(n)$ to calculate predictions $\hat{x}(n+1)$ and $\hat{y}(n+1)$ for the following input sample in the encoder and decoder, respectively. Obviously, the reconstructed signals in encoder and in decoder are equal $\tilde{x}(n) = y(n)$, if no transmission error occurs. As the predictor in encoder and in decoder operates with the same signal values, it is not necessary to transmit the filter coefficients.

A FIR filter in lattice structure [10], [11] is used for the prediction calculation. The block diagram of a pth-order prediction error filter in lattice structure is shown in Fig. 2. The signals $f_m(n)$ and $b_m(n)$, where $m = 0, \ldots, p$ and p denotes prediction order, are used to compute the prediction error $e(n)$. The signals $f_m(n)$ and $b_m(n)$ at lattice stage m are recursively obtained by

$$f_m(n) = f_{m-1}(n) - k_m b_{m-1}(n-1) \tag{1}$$

$$b_m(n) = b_{m-1}(n-1) - k_m f_{m-1}(n), \tag{2}$$

where the output $f_p(n)$ is the prediction error with the LPC filter input $f_0(n) = b_0(n) = \tilde{x}(n)$. The desired prediction is computed by $\hat{x}(n) = \tilde{x}(n) - f_p(n)$.

To obtain adaptive filter coefficients $k_m(n)$, which are used for the prediction error minimization, the gradient adaptive lattice (GAL) algorithm [12] is applied. The coefficients are updated iteratively according to

$$\begin{aligned} k_m(n+1) &= k_m(n) + \\ &\mu_m(n) \cdot \big(f_m(n) b_{m-1}(n-1) + b_m(n) f_{m-1}(n)\big). \end{aligned} \tag{3}$$

The gradient weights $\mu_m(n)$ are calculated for every lattice stage by normalizing the base gradient weight $\tilde{\mu}$ by the energy of signals $f_{m-1}(n)$ and $b_{m-1}(n)$ from the previous lattice stage [13], [14]. The gradient weights are given by

$$\mu_m(n) = \frac{\tilde{\mu}}{\sigma_m^2(n) + \sigma_{min}^2} \tag{4}$$

$$\sigma_m^2(n) = (1-\tilde{\mu})\sigma_m^2(n-1) + \tilde{\mu}\big(f_{m-1}^2(n) + b_{m-1}^2(n)\big), \tag{5}$$

where σ_{min}^2 is a small constant to avoid division by zero.

The important property of the lattice filter structure is the minimum phase property, so that the filter stability can be guaranteed by limiting the filter coefficients to $|k_m(n)| < 1$. The GAL algorithm is chosen due to its low computational complexity.

B. Block companding

Now we present a design description of the block compander for employing it in the DPCM coding scheme. First, the signal $x(n)$ which has to be quantized is divided into blocks of fixed length M. The introduced algorithmic delay is proportional to the chosen block length M. For example,

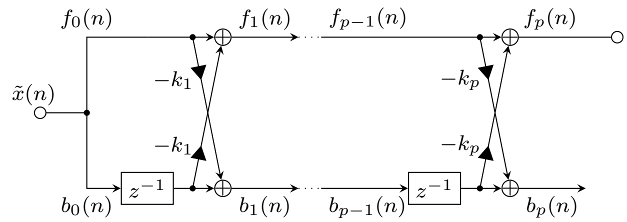


Fig. 2. Prediction error filter in the lattice structure.

$M = 16$ leads to a delay of 0.36 ms at a sampling frequency of 44.1 kHz. Next, the absolute maximal value $x_{max}(k)$ is determined for every block, where k is the current block number. The values in the block k are normalized by the scaling factor $x_{max}(k)$. The normalized values $q(i) = \frac{x(i)}{x_{max}(k)}$, where $i = n-M, n-(M-1), \ldots, n$ is a time index of the current block, are fed to the quantizer. As the values in the block are normalized, no clipping occurs during the quantization process, if the scaling factor $x_{max}(k)$ stays unaltered.

For the reconstruction of the quantized signal in the block k, the scaling factor $x_{max}(k)$ has to be transmitted to the decoder as side information. The quantization scheme for the scaling factor relies on the assumption that, as the lattice prediction filter is stable, the range of maximal absolute values in the block is limited by $10^{-10} \leq x_{max}(k) \leq 1$. The lower bound is a non-zero value to avoid division by zero during the block normalization process. To prevent big distortions during quantization of small valued scaling factors, the same range is logarithmically represented as $-200 \leq 20\log_{10}(x_{max}(k)) \leq 0$ which defines a unipolar quantizer for the coding of $x_{max_{log}}(k)$. To avoid clipping of the values inside the block k, a new scaling factor $x'_{max}(k)$ has to be chosen so that

$$\begin{aligned} x'_{max}(k) \overset{\approx}{\geq} \tilde{x}_{max}(k) \qquad &\text{with} \\ \tilde{x}_{max}(k) = \mathrm{Q}^{-1}\Big(\mathrm{Q}&\left(x_{max}(k)\right)\Big), \end{aligned} \tag{6}$$

where $\mathrm{Q}(\cdot)$ and $\mathrm{Q}^{-1}(\cdot)$ are the quantization and the reconstruction operation, respectively.

The overhead per sample is inversely proportional to the block length M. For example, $M = 16$ and a word length of $w_{x_{max}} = 6$ for the bit representation of the scaling factor $x_{max_{log}}(k)$ would lead to an overhead of 0.375 bit/sample. Setting the block length M is deciding between the amount of algorithmic delay, the overhead size per sample and, as shown in the results section III, the audio quality.

C. Block companded DPCM

A block companding technique is often applied in subband coding schemes [15]. The straight replacement of the fixed quantizer in the encoder of the DPCM coding scheme in Fig. 1 by the block companded quantizer is impossible due the sample-wise quantization and reconstruction of prediction error $\tilde{e}(n)$ which is fed back to the predictor P to calculate successive prediction. To determine the scaling factor $x_{max}(k)$, all M samples of the block have to be present. This makes sample-wise encoding unfeasible.

The concept of block companded DPCM (BCDPCM) which allows sample-wise processing can be summarized as follows:

1) Block start. Calculate prediction error $e(n)$. If $x_{max}(k)$ is not present or $x_{max}(k) < |e(n)|$, set $x_{max}(k) = |e(n)|$ and proceed with next sample in the next step.
2) Based on the absolute value of the current prediction error $|e(n)|$ in the block, several cases occur:
   - If $|e(n)| \leq x_{max}(k)$, continue with coding next sample till whole block of M samples is present.
   - If at any position in the block $|e(n)| > x_{max}(k)$ and the block recalculation limit is not reached, increase block recalculation counter and update $x_{max}(k) = |e(n)|$. Leave this step and go to block start to proceed with the step 1.
   - If $|e(n)| > x_{max}(k)$ and the recalculation limit is reached, use current $x_{max}(k)$ for the calculation of all following values in the block k and continue with the next step.
   - The $x_{max}(k)$ is determined if the last sample in the block is reached. Go to the next step.
3) First transfer $x_{max}(k)$ to the decoder and then transfer sample-wise coded block samples. Since for the decoder $x_{max}(k)$ is present, the reconstruction of $y(n)$ is possible sample-wise.

To ensure that at third step in the summary the same $x_{max}(k)$ for coding and decoding is used, in the first two steps $x_{max}(k)$ has to be quantized and reconstructed as described in the previous section II-B of the block compander design.

The determination of the scaling factor in the first and second step of the summary is computationally expensive. Including a preprocessing step to the scaling factor computation allows to reduce complexity:

- Calculate the prediction error for every block sample $e(n)$ and update $x_{max_{pre}}(k) = |e(n)|$ for quantization of $e(n)$ if $x_{max_{pre}}(k) < |e(n)|$. Proceed till block k end is reached and predetermined $x_{max_{pre}}(k)$ is present. Set $x_{max}(k) = x_{max_{pre}}(k)$ and continue with step 1 from the summary of the scaling factor determination.

As shown in the simulation results of section III, nearly the same audio quality can be reached using the coding scheme with the preprocessing step as without preprocessing.

III. Simulation results

For the simulation of the audio quality dependence of the block length by using the proposed BCDPCM coding scheme, the selected test mono signals from SQAM CD set [16] are used. The SQAM audio excerpts (starting form 0.5 s and 10 s long) with sampling frequency 44.1 kHz are coded using the word length of 3 bit/sample and 6 bit/$x_{max}(k)$ for payload and scaling factor quantization, respectively. The BCDPCM test parameters are set as follows: the predictor order $p = 32$, the base gradient weight $\tilde{\mu} = 2.08112 \cdot 10^{-3}$, minimal energy of lattice stage error signals $\sigma^2_{min} = 1.46494 \cdot 10^{-5}$. The payload quantizer with the test codebook and partition parameters is given in Table I and the scaling factor quantizer remains

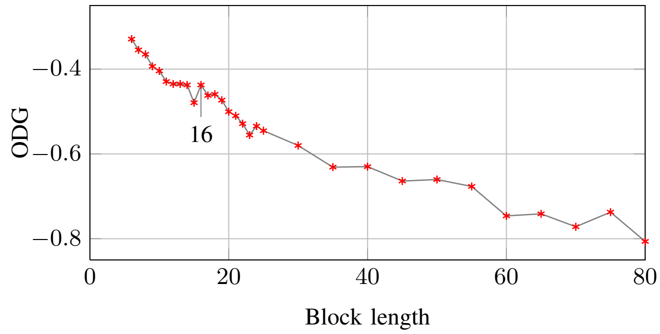


Fig. 3. BCDPCM mean quality dependence of block length. Stars mark simulation points at given block length starting from $M = 6$.

TABLE I
BCDPCM payload 3 bit quantizer parameters.

Partitions	Codebook
−0.7464	−0.8676
−0.5133	−0.6252
−0.2976	−0.4015
−0.0969	−0.1938
0.1248	0
0.3869	0.2497
0.6765	0.5241
	0.8288

the same as in II-B section, assuming that a noise free channel is used. The audio quality of the coded test signals are compared to the reference signals based on the ITU-R BS.1387-1 (PEAQ) method [17], [18]. The method classifies the perceptual audio quality by objective difference grades (ODG) on a scale from -4 (very annoying) to 0 (imperceptible). The simulation result is shown in Fig. 3. The first simulation point with block length $M = 6$ leads to the best quality and to the shortest delay, but the resulting overhead 1 bit/sample is too high. Starting from the block length $M = 16$ the mean audio quality over tracks starts to decrease rapidly. Because of the small introduced delay 0.36 ms, measured good mean quality of -0.4373 and reasonably small overhead per sample of 0.375 bit/sample, the block length parameter $M = 16$ is used for the simulations that follow.

The results of the BCDPCM coding scheme without and with the preprocessing step for selected tracks are shown in Fig. 4. The maximal number of block recalculations, which is needed to determine the scaling factor is limited to 11 and to 6 for coding without and with preprocessing step, respectively. Increasing the maximal recalculation number for both methods does not affect the audio quality. This simulation demonstrates that nearly the same audio quality can be reached using the preprocessing step which significantly reduces the computational complexity of the BCDPCM coding scheme. The mean quality difference between the proposed coding schemes without and with preprocessing step over all SQAM tracks is 0.02 in ODG-scale. The comparison of several signals coded by the BCDPCM coding scheme without preprocessing step and the plain ADPCM proposed with optimal

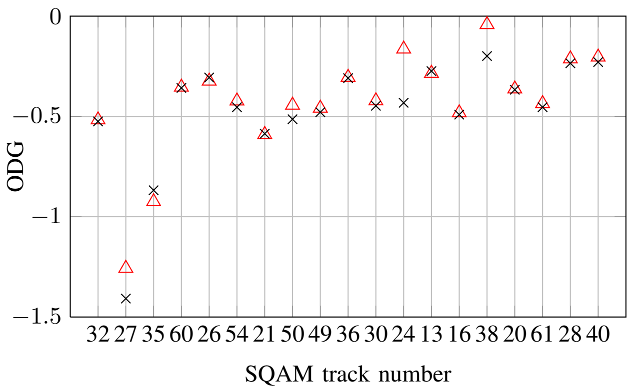


Fig. 4. Comparison of BCDPCM without preprocessing step by limiting total block recalculations to 11 (triangles) and with preprocessing by step limiting recalculations to 6 (crosses).

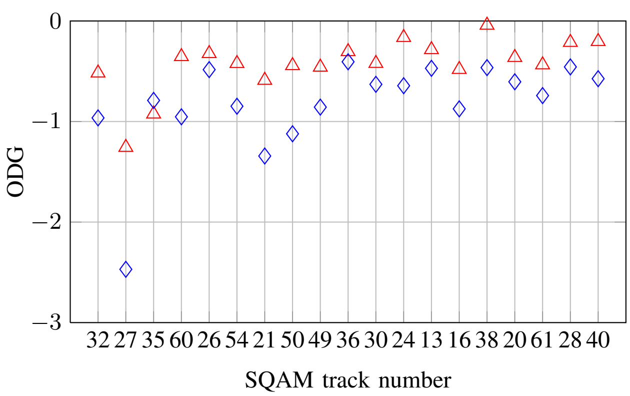


Fig. 5. Comparison of BCDPCM (triangles) and plain ADPCM (diamonds) audio coding schemes using word length of 3.375 bits/sample and 3 bits/sample, respectively.

parameters in [14] is shown in Fig. 5. As expected, the signals coded with BCDPCM show an improvement in the audio quality compared to ADPCM coded signals. The track 35 glockenspiel shows a light quality degradation compared to ADPCM coded signal and the coded track 36 xylophone exhibits similar quality. In those cases the similar quality is caused by a slowly decaying prediction error, as the adaptive quantizer in the ADPCM is able to follow the envelope of the residual signal and the BCDPCM does not exhibit quality improvement. The best perceptual quality improvement shows the track 27 castanets, where short transient signal periods produce transient-like prediction error and the audio quality degrades using an ADPCM coding scheme. In the mean over all SQAM track set, the proposed system with -0.3875 in ODG-scale achieves significantly better audio quality than with plain ADPCM method coded signals with -0.7346 in ODG-scale. The mean perceptual audio quality improvement is 0.3471 in ODG-scale.

IV. Conclusions

We have proposed a new coding scheme BCDPCM which incorporates block companded quantization with differential pulse code modulation. The proposed system introduces a small delay to the coding scheme due to the scaling factor determination on the encoder side and also adds an overhead. The determination of the scaling factor leads to a more complex coding scheme, but the proposed BCDPCM provides significantly better audio quality compared to the plain ADPCM coding scheme. We believe that further sound quality improvements are possible by adding noise shaping techniques to the proposed BCDPCM coding scheme and by optimizing coding parameters.

References

[1] Gayer Marc, Lutzky Manfred, Schuller Gerald, Krämer Ulrich, and Wabnik Stefan, "A guideline to audio codec delay," in *Audio Engineering Society Convention 116*, May 2004.

[2] Aki Härmä and Unto K. Laine, "Warped low-delay CELP for wideband audio coding," in *Audio Engineering Society Conference: 17th International Conference: High-Quality Audio Coding*, Aug. 1999.

[3] B. S. Atal and M. R. Schroeder, "Adaptive predictive coding of speach signals," 1970, The Bell System Technical Journal.

[4] Jerry D. Gibson, "Sequentially adaptive backward prediction in ADPCM speech coders," Jan. 1978, vol. 26, pp. 145 – 150.

[5] Martin Holters and Udo Zölzer, "Delay-free lossy audio coding using shelving pre- and post-filters," in *Acoustics, Speech and Signal Processing. ICASSP 2008. IEEE International Conference on*, Apr. 2008, pp. 209 – 212.

[6] K. Niwa, T. Araseki, and A. Tomozawa, "A new channel bank with block companding," *Communications, IEEE Transactions on*, vol. 30, no. 4, pp. 574 – 580, Apr. 1982.

[7] D. Cohn and J. Melsa, "The residual encoder – an improved ADPCM system for speech digitization," Sep. 1975, vol. 23, pp. 935 – 941.

[8] International Telecommunication Union, "32 kbit/s adaptive differential pulse code modulation (ADPCM). ITU-T recommendation G.721," 1988.

[9] Martin Holters, Oliver Pabst, and Udo Zölzer, "ADPCM with adaptive pre- and post-filtering for delay-free audio coding," in *Acoustics, Speech and Signal Processing, 2007. ICASSP 2007. IEEE International Conference on*, Apr. 2007, vol. 1, pp. I–269 –I–272.

[10] J. Makhoul, "Stable and efficient lattice methods for linear prediction," Oktober 1977, vol. 25, pp. 423 – 428.

[11] R. Reininger and J. Gibson, "Backward adaptive lattice and transversal predictors for ADPCM," in *Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP '84.*, Mar. 1984, vol. 9, pp. 429 – 432.

[12] L. Griffiths, "A continuously-adaptive filter implemented as a lattice structure," in *Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP '77.*, May 1977, vol. 2, pp. 683 – 686.

[13] C. Gibson and S. Haykin, "Learning characteristics of adaptive lattice filtering algorithms," *Acoustics, Speech and Signal Processing, IEEE Transactions on*, vol. 28, no. 6, pp. 681 – 691, dec 1980.

[14] Martin Holters, Christian R. Helmrich, and Udo Zölzer, "Delay-free audio coding based on ADPCM and error feedback," in *Proc. of the 11th Int. Conference on Digital Audio Effects (DAFx-08)*, Sep. 2008.

[15] D. Esteban and C. Galand, "Application of quadrature mirror filters to split band voice coding schemes," in *Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP '77.*, May 1977, vol. 2, pp. 191 – 195.

[16] Belgium European Broadcasting Union, Bruxelles, "Sound quality assessment material: Recordings for subjective tests – CD and user's handbook for the EBU-SQAM compact disc," April 1988, Technical Centre of the EBU.

[17] International Telecommunication Union, "ITU recommendation ITU-R BS.1387-1, method for objective measurements of perceived audio quality (PEAQ)," November 2001.

[18] P. Kabal, "An examination and interpretation of ITU-R BS.1387: Perceptual evaluation of audio quality," 2002, McGill University, Telecommunications & Signal Processing Laboratory.