LOW-DELAY VECTOR-QUANTIZED SUBBAND ADPCM CODING

Marco Fink, Udo Zölzer

Department of Signal Processing and Communications,
Helmut-Schmidt University Hamburg
Hamburg, Germany
marco.fink@hsu-hh.de

ABSTRACT

Several modern applications require audio encoders featuring low data rate and lowest delays. In terms of delay, *Adaptive Differential Pulse Code Modulation* (ADPCM) encoders are advantageous compared to block-based codecs due to their instantaneous output and therefore preferred in time-critical applications. If the the audio signal transport is done block-wise anyways, as in Audio over IP (AoIP) scenarios, additional advantages can be expected from block-wise coding. In this study, a generalized subband ADPCM concept using vector quantization with multiple realizations and configurations is shown. Additionally, a way of optimizing the codec parameters is derived. The results show that for the cost of small algorithmic delays the data rate of ADPCM can be significantly reduced while obtaining a similar or slightly increased perceptual quality. The largest algorithmic delay of about 1 ms at 44.1 kHz is still smaller than the ones of well-known low-delay codecs.

1. INTRODUCTION

A major part of today's communication is based on packet-switched networks and especially IP-networks. The requirement for data rate compression techniques to transmit audio signals with a reduced data rate but good quality emerged in the very beginning of digital audio transmission and led to development of many specialized audio codecs, which are utilized in several applications. However, certain applications with a strong interactive characteristic, like wireless digital microphones, in-ear monitoring, online gaming, or Networked Music Performances (NMP) [1, 2], define the requirement of very small delays additionally. Several coding approaches, like the ultra-low delay codec (ULD) [3, 4], the aptX® codec family, and the *Constrained-Energy Lapped Transform* (CELT) part of the OPUS codec [5, 6], were presented to fulfill the additional low-delay requirement.

This study shall reveal the performance gain when a broadband ADPCM codec, as described in [7, 8, 9], is operating in subbands by extending it with a filterbank. Additionally, the scalar quantizer is replaced with a vector quantizer to create a vector-quantized subband ADPCM (VQSB-ADPCM). In contrast to the transform-based OPUS codec, the ULD and aptX® are also ADPCM codecs. However, the ultra-low delay codec is a broadband approach and an open loop implementation. In other words, no error feedback is used. Specific system differences to the aptX® codec family can't be asserted due to missing documentation. A subband ADPCM coding structure with a vector quantization was also proposed in [10]. However, the authors utilized known ADPCM predictors of the G.721 standard whereas in this work a prediction based on lattice filters, optimized to the proposed coding structure is used. In addition, the comparability of the codec from [10] is unfeasible due to non-compliant test methods and material. Robustness techniques against bit errors for ADPCM codecs as shown in [11] are not considered in this study.

This study is structured as follows. The proposed codec structure with all its modules is explained in Sec. 2, whereas the undertaken codec parameter optimization is explained in Sec. 3. The evaluation with automated measurements is presented in Sec. 4. Sec. 5 summarizes the findings of this study and depicts possible enhancements of the proposed codec structure.

2. SYSTEM OVERVIEW

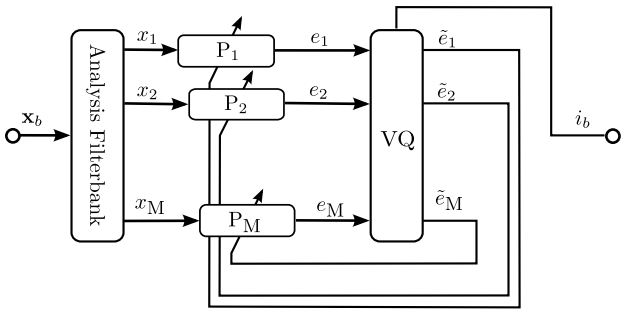


Figure 1: *Blocks scheme of encoder*

The proposed encoder is constructed block-wise as shown in Fig. 1. The b_{th} input signal block $\mathbf{x}_\text{b}$ is fed to an analysis filter bank, resulting in M critically sampled subband signals, denoted as $x_1(b), \ldots, x_M(b)$. M is the number of bands and the block size of $\mathbf{x}_\text{b}$. Every subband signal $x_1(b), \ldots, x_M(b)$ is individually processed by a distinct ADPCM encoder producing a residual signal $e_1(b), \ldots, e_M(b)$, which is massively reduced in entropy and amplitude by prediction. In a final step, a vector quantizer is applied to find a codebook entry $\mathbf{c}_b$ with index i_b describing the residual signal vector $\mathbf{e}(b) = [e_1(b), \ldots, e_M(b)]$ best.

At the decoder side (Fig. 2), the residual vector $\mathbf{e}$ is regained through a lookup operation in a codebook using i_b. The decoder uses the same predictors as the encoder to reconstruct the subband signals $\tilde{y}_1(b), \ldots, \tilde{y}_M(b)$. To reproduce a full-band signal with the original sampling a synthesis filter bank is used, which produces the final output block $\mathbf{y}_\text{b}$, serialized in the continuous output signal $y(n)$.

The signals within the encoder are illustrated in Fig. 3. A monophonic tune played on a classical concert guitar is the exemplary input signal $x(n)$ in Fig. 3a). The corresponding fil-

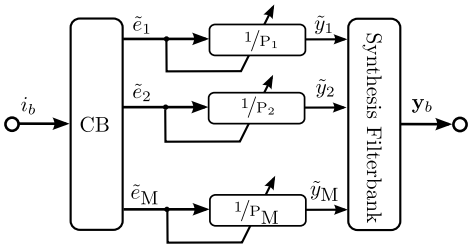


Figure 2: *Block scheme of decoder*

terbank output $x_1(b), \ldots, x_M(b)$ for $M = 8$ in Fig. 3b) shows that for stationary musical signals the energy of the input signal is mainly concentrated in the lower bands. Transient components like the plucking tend to show more energy in higher bands. The redundancy-reduced residual signals $e_1(b), \ldots, e_M(b)$ and the quantized residual signals $\tilde{e}_1(b), \ldots, \tilde{e}_M(b)$ are depicted in Fig. 3c) and Fig. 3d), respectively. Apparently, they can't be graphically distinguished when a 16 bit codebook is used. The corresponding codebook index $i(b)$ is plotted in Fig. 3e). When the codebook is sorted in order of ascending l_2-norm of the codebook entries $||\tilde{\mathbf{e}}||$, $i(b)$ tends to be small and hence the *most significant bit* (MSB) is mainly unused in contrast to the *least significant bit* (LSB).

All functional modules of the codec structure are explained in greater detail in the following.

2.1. Filter Bank

The authors considered two critically-sampled (nearly) perfect reconstruction filter banks:

2.1.1. Cosine-modulated FIR filter bank

A nearly perfect-reconstruction FIR filter bank was designed similar to [12]. First, a prototype filter $H_{\text{PT}}(e^{j\Omega})$ with a -3 dB cutoff at $\Omega = \frac{\pi}{M}$ is created iteratively using the window method with a Hamming window. Then its impulse response $h_{\text{PT}}(n)$ of length N is modulated for all subbands $m = 1, \ldots, M$

$$h_m(n) = 2 \cdot h_{\text{PT}}(n) \cdot \cos\left((-1)^m \tfrac{\pi}{4} + \tfrac{\pi}{M}(m + \tfrac{1}{2})(n - \tfrac{N-1}{2})\right) \quad (1)$$

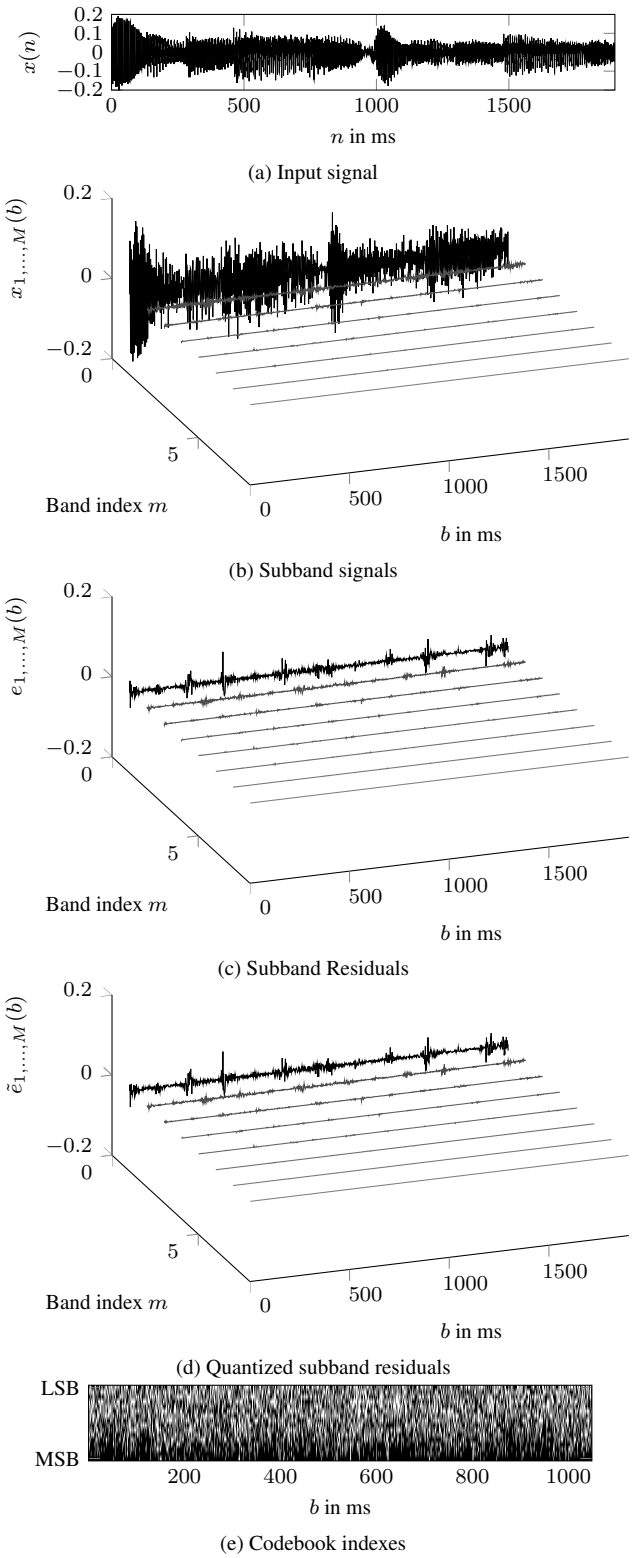
to obtain the impulse responses for the analysis filter bank. The corresponding synthesis filter bank impulse responses are simply time-reversed

$$g_m(n) = h_m(N - n). \quad (2)$$

The delay of a single FIR filter featuring a symmetric impulse response is known to be $\frac{N-1}{2}$ and hence the overall delay is $d_{\text{FIR}} = N - 1$. Since the study examines low-delay codecs the impulse response should be as small as possible to show a small delay. On the other hand, the side band attenuation scales with the filter length N.

2.1.2. Modified Discrete Cosine Transform (MDCT)

In addition to the FIR filterbank, a MDCT was chosen due to its well-known advantageous properties like low-frequency energy compression and time-domain aliasing cancellation (TDAC). Since it is a half-overlapped transform and applied twice, its algorithmic



(a) Input signal

(b) Subband signals

(c) Subband Residuals

(d) Quantized subband residuals

(e) Codebook indexes

Figure 3: *Signals throughout the encoder for a plucked nylon guitar sequence*

delay $d_{\text{MDCT}} = M$ is equal to the amount of bands. The impulse response for the m_{th} band is defined as

$$h_m(n) = \sqrt{\frac{2}{M}} \cdot w(n) \cdot \cos\left(\frac{(2n+M+1)(2k+1)\pi}{4m}\right). \quad (3)$$

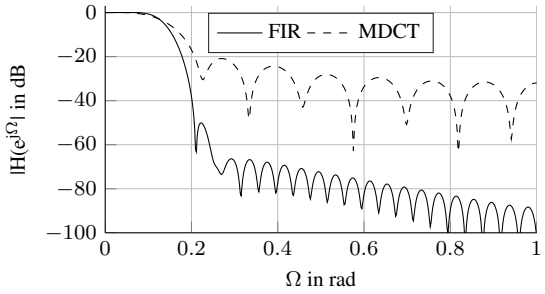


Figure 4: *Band $m = 1$ of FIR and MDCT filterbank for $M = 8$, $N = 51$*

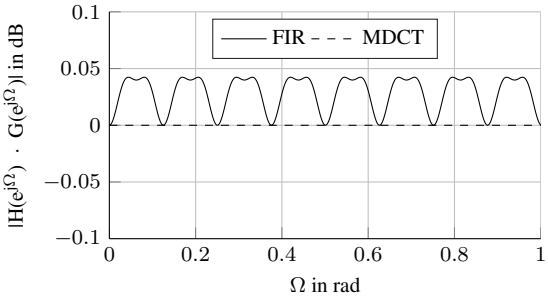


Figure 5: *Reconstruction error of FIR and MDCT filterbank for $M = 8$, $N = 51$*

Comparing the absolute frequency responses $|H_{\text{MDCT}}(e^{j\Omega})|$ and $|H_{\text{FIR}}(e^{j\Omega})|$ in Fig. 4 reveals that the stopband attenuation of the MDCT is significantly worse than in the FIR case. The reconstruction error of both filterbanks are depicted in Fig. 5. In contrast to the MDCT filterbank, the reconstruction of the FIR filterbank deviates up to $0.045\,\text{dB}$ from the perfect MDCT reconstruction. As demonstrated in [13], the design of a perfectly reconstructing FIR filterbank is possible when the impulse response length is restricted to a length of $2 \cdot M \cdot K$, where K is a positive integer. Nevertheless, the stopband attenuation is of predominant importance for the coding efficiency as derived later on.

2.2. Prediction

As previously mentioned, the idea of an ADPCM codec is to solely quantize prediction residuals. Therefore, M predictors $\text{P}_{1,\ldots,M}$ are utilized in every band to estimate the following sample $\hat{x}_m$ from prior values of x_m. Only the subband prediction error $e_m = x_m - \hat{x}_m$ is fed to the following quantization stage. At the decoder side, the same predictor the encoder utilizes is applied to predict the signal $\hat{y}_m$. The transmitted decoded prediction error signal $\tilde{e}_m$ is added to obtain the decoded subband signal $\tilde{y}_m$. To allow various signal conditions, the predictors filters are adapted using the *Gradient Adaptive Lattice* (GAL) technique [14]. To ensure a synchronous prediction filter adaption in encoder and decoder, the quantized error signal $\tilde{e}_m$ is utilized for the adaption step instead of the original error signal e_m.

The adaption of lattice filter coefficients is adaptive since the gradient weight of the lattice stage p

$$\mu_p(n) = \frac{\lambda}{\sigma_p(n) + v_{\text{min}}} \quad (4)$$

is computed using a base step size λ and normalizing it with an instantaneous power estimate $\sigma_p(n)$ at time instance n. v_{min} is a small constant avoiding a division by zero. The power estimate $\sigma_p(n)$ is computed recursively

$$\sigma_p(n) = (1-\lambda)\,\sigma_p(n) + \lambda\,(f_{p-1}^2(n-1) + b_{p-1}^2(n-1)), \quad (5)$$

where $f_{p-1}(n-1)$ and $b_{p-1}(n-1)$ are the forward- and backward prediction error of the previous lattice stage at the previous time instance.

Since the signal characteristics differ drastically throughout the bands, they have to be individually adjusted. In this study, the prediction order N_m^p, base step size λ_m, and v_m of the filter adaption are carefully adjusted as described in Sec. 3.

2.3. Adaptive Vector Quantization

The process of vector quantization is defined as the mapping of an input vector to an equal-sized output vector, taken from a finite-sized codebook, resulting in the smallest distortion between the vectors. Due to the harmonic characteristics of sound and the imperfections of the applied filterbanks significant correlation between the subbands occur. Hence, individual scalar quantization would result in higher distortion than applying a vector quantization which can exploit the joint probability density function of the subbands. The euclidean distance between the current prediction error residual vector $\mathbf{e}$ and all codewords $\mathbf{C}_i$ in a codebook is minimized

$$\min_i \quad \left[(\mathbf{e} - \mathbf{C}_i)^T (\mathbf{e} - \mathbf{C}_i)\right] \quad (6)$$

to find the best-fitting match from the codebook and only the index i of the codebook table is transmitted.

The quantization is made adaptive by computing normalized subband residuals

$$\bar{e}_m = \frac{e_m}{v_m} \quad (7)$$

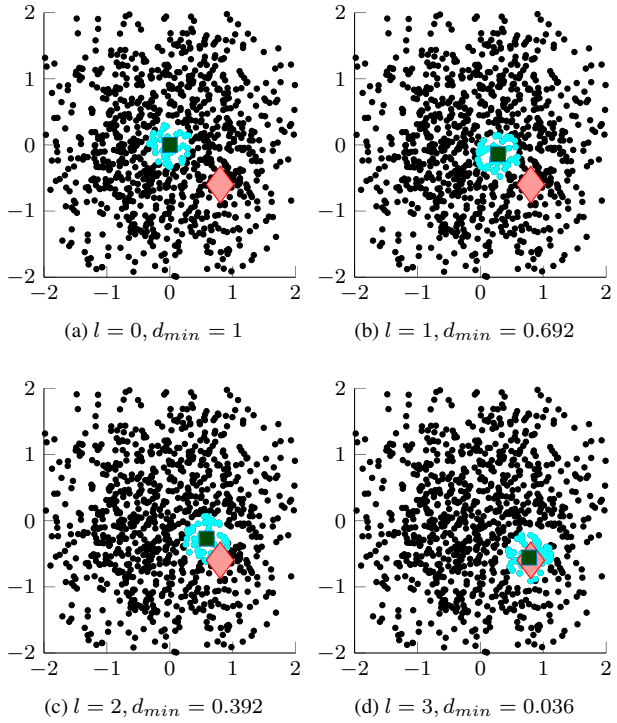
by dividing them by their recursively estimated envelope v_m to achieve a nearly constant signal variance [15] as described in [9]. Note that the computation the envelope estimation is signal-adaptive since different smoothing coefficients λ_{AT} and λ_{RT} are utilized for the attack and release case. Additionally, values of v_m smaller than v_{min} are clipped to v_{min}. Similar to [10] the distance function can be weighted

$$\min_i \quad \left[(\mathbf{e} - \mathbf{C}_i)^T \text{diag}(\mathbf{w})(\mathbf{e} - \mathbf{C}_i)\right] \quad (8)$$

using a weighting vector $\mathbf{w}$ to emphasize certain bands during the codebook search. In this study, the envelope estimation v_m is utilized to define $\mathbf{w} = v_1, \ldots, v_M$.

The codebook was designed using normally-distributed noise. The codebook size yields $M \times 2^{R_b M}$ entries. For $R_b = 2$ bits per sample and $M = 8$ subbands this corresponds to 8×65536 entries. Since a linear search in large codebooks is very costly, *Nearest*

Neighbour Search (NNS) was implemented. For that purpose, a second table $\mathbf{N_n}$ of size $K \times 2^{R_b M}$ containing the K indexes of the codewords with smallest euclidean distance has to be saved for every codeword.



(a) $l = 0, d_{min} = 1$ (b) $l = 1, d_{min} = 0.692$

(c) $l = 2, d_{min} = 0.392$ (d) $l = 3, d_{min} = 0.036$

Figure 6: *Illustration of the NNS search procedure for the 2D case and* 3 *iterations*

The search process is notated in Alg. 1 and illustrated in Fig. 6 for the 2D case, a codebook size of 2×1024 and $K = 50$. Initially, the euclidean distance between the normalized residual vector $\bar{\mathbf{e}}$ (marked in red) and the first entry of the codebook $\mathbf{C}_{i_{min}}$ (green), representing silence, is computed and set to the initial minimum distance d_{min}. Then the euclidean distance of all K nearest neighbours (cyan) to $\bar{\mathbf{e}}$ is computed and saved in $d(k)$. d_{min} is replaced by $d(k)$'s smallest value d_{nn} if it is smaller. Otherwise the search method terminates. This procedure is repeated $\mathrm{l}_{\max}$ times. Examining Fig. 6 shows, that the cloud of nearest neighbours is moving progressively in the direction of $\bar{\mathbf{e}}$. The codebook entry showing the smallest distance $d_{min} = 0.036$ is already found in the third iteration in this example and hence, only $4 \cdot 50 = 200$ computations of the vector distance had to be computed instead of 1024 for the linear codebook search.

Algorithm 1 *Nearest neighbor search*

```
i_min = 1
d_min = [(e − C_{i_min})^T diag(w)(e − C_{i_min})]
for l = [1, ..., l_max] do
    for k = [1, ..., K] do
        i = N_n(k, i_min)
        d(k) = [(e − C_i)^T diag(w)(e − C_i)]
    end for
    [d_nn, i_nn] = min(d)
    if d_nn > d_min then
        break
    else
        d_min = d_nn
        i_min = i_nn
    end if
end for
```

3. OPTIMIZATION

As mentioned before, a variety of prediction parameters have to be optimized for this codec structure design. The optimization of codec parameters was done in a two-step approach and is restricted to a filterbank size of $M = 8$ in this study. At first, the authors wanted to roughly identify the order N_m^p and base step λ_m for all bands. For every band m, a chosen set of orders N_m^p, and both filterbanks a gradient descent algorithm is applied to find the base step size λ_m by minimizing the cost function

$$C_{\mathrm{gd}} = \frac{1}{F} \sum_{f=1}^{F} \sum_{n=0}^{N_f - 1} e_m^2(n) \tag{9}$$

describing the accumulated prediction error energy within a band m averaged over a large data set consisting of F tracks with a length of N_f samples, respectively. The actual computation of e_m was performed with a real-time C implementation of the encoder with disabled quantization. The utilized data set is the *Sound Quality Assessment Material* (SQAM) [16] dataset, consisting of 70 stereo tracks incorporating lots of different signal sources and mixes, which are recorded with a sampling rate f_s of 44.1 kHz. For all following evaluations the tracks were downmixed to a single channel.

In a second step, the optimization using simulated annealing using a perceptually motivated cost function similar to [17] was performed. Perceptual evaluation of audio quality is a tricky task due to the subjectivity of the nature of human hearing. Nevertheless, the *Perceptual Evaluation of Audio Quality* (PEAQ) [18] algorithm is established as standard tool for this purpose. A self-implemented tool, verified in [19], is fed with an original wavefile and a wavefile processed with encoder and decoder. The result of the PEAQ algorithm is the so-called *Objective Difference Grade*, ranging from -4 to 0 corresponding to the measured impact of coding artifacts describing the impairment as "very annoying" (-4) to "imperceptible" (0). The cost function

$$C_{\mathrm{ODG}} = \frac{1}{F} \sum_{f=1}^{F} \mathrm{ODG}_f^4 \tag{10}$$

emphasizes negative ODG scores since the fourth power of ODG scores is used. Hence, an optimization towards a globally good audio quality instead of excellent quality for many tracks but strong negative outliers can be expected. Additionally to N_m^p and λ_m, the minimum envelope amplitude $v_{\min}$, and the smoothing coefficients for the envelope estimation λ_{AT},λ_{RT} are optimized in a second step.

The results of the second optimization step are depicted in Tab. 1. The trend of the prediction order N_m^p over the bands m, shown in Tab. 1a), falls from about 120 to 20 for the MDCT and FIR case. In contrast, the base step sizes λ_m (Tab. 1b)) differ drastically for both filterbanks. While λ_m shows a concave trend ranging from 0.08 to 0.01 and back to 0.09 for FIR, the MDCT λ_m are

Table 1: *Overview of optimization results*

	(a) N_m^p		(b) $\lambda_m \cdot 1e^{-2}$		(c) $v_{\min} \cdot 1e^{-4}$	
Band m	FIR	MDCT	FIR	MDCT	FIR	MDCT
1	119	127	0.7945	0.4247	0.1182	0.1420
2	112	133	0.5950	0.9517	0.8280	0.0835
3	88	60	0.6225	0.9687	0.4174	0.0067
4	75	80	0.3590	0.9468	1.1363	0.1671
5	42	55	0.3138	0.9987	1.0234	0.1678
6	27	20	0.1339	0.9717	1.2486	0.0757
7	27	16	0.2826	0.9319	0.1754	0.1727
8	19	30	0.8991	0.9934	0.0052	0.0577

nearly constant 0.09 except for 0.04 at $m = 1$. Also the results for $v_{\min}$ in Tab. 1c) are very distinguishable. For $m = 2, \ldots, 6$ the minimum envelope amplitude is significantly higher in the FIR case. Unexpectedly, the envelope smoothing coefficients were optimized to very similar values ($\lambda_{\text{AT}} = 0.8, \lambda_{\text{RT}} = 0.1$) as in the broadband case [17] for both filterbanks.

4. EVALUATION

Using the optimized codec parameters, the evaluation of the codec structure can be discussed in the following. The evaluation is established using the PEAQ tool and the SQAM dataset as before. The proposed codec structured is compared to the optimized single-band ADPCM codec without noise shaping from [17] using $R_b = 3$ bits per sample, corresponding to 132.3 kbit/s for a sampling frequency $f_s = 44100$ Hz. The VQSB-ADPCM is configured with $R_b = 2$ bits, equivalent to 88.2 kbit/s, and hence every residual vector $\bar{\mathbf{e}}$ is encoded using 16 bit. A comparison with the same bitrate is not directly feasible due to the resulting huge codebook of size $8 \times 2^{3 \cdot 8}$. The ODG scores for the 3-bit reference and several codec configurations are shown in Fig. 7. The reference broadband codec achieves ODG scores in the range of $[-1.4, -0.075]$ and an average score of -0.5, indicating good quality throughout the test items. Only the castanets (27) and the accordion (42) sample produce scores below -1. The ODG scores for the VQSB-ADPCM with the FIR filterbank are similar to the reference codec for most items. Items (2-6), which are synthesized gong signals, horn (23), tuba (24), castanets (27), vibraphone (37), tenor (46), bass (47) show a massive degradation in audio quality in contrast to the reference. Apparently, low-pitched and percussive signals benefit from broadband encoding. Neglecting item (1-7) due to their minor importance for the applications mentioned in Sec. 1, the average ODG score constitutes -0.73. In other words, the outcome of the proposed codec structure is a relative bit rate reduction to $2/3$ for the cost of 0.24 mean ODG score degradation. However, the standard deviation is rising from 0.21 to 0.50, indicating a larger variance of the determined audio quality.

Utilizing the MDCT filter bank and the related codec parameter performed significantly worse as apparent from Fig. 7. The scores for the FIR case can never be reached for the relevant items and is even less consistent. The average score of -2.54 clearly discloses the missing usability of the codec in this configuration.

So far, only the results for the linear codebook search were considered. The implemented NNS codebook search is not guaranteed to find the globally best codeword from the codebook and hence a degradation of performance in exchange of decreased computation time has to be expected. Fortunately, the curves for linear and NNS search only differ slightly throughout the test set except for bass (47) item. For $K = 100$ nearest neighbours, a mean deviation of 0.01 and 0.16 is identified for the FIR and MDCT case, respectively. The averaged computation time for encoding the SQAM test set with a mean item duration of 50 s with the non-optimized C implementation on a Desktop PC with i5-3570K CPU clocked with 3.4 GHz is depicted in Fig. 8. The broadband ADPCM reference required about 1.44 s and hence, is approximately 35 times faster than realtime. Applying the filterbank and the linear codebook search increases the encoding time up to 315.23 and 320.81 s for FIR and MDCT configuration, respectively. This results in an impracticable performance of 0.15 times realtime. Applying NNS decreases the encoding drastically to 7.02 and 7.52 s. The decoding times are clearly smaller since the inverse quantization is a simple table lookup operation. The reference broadband decoder required 1.30 s, whereas the MDCT and FIR VQSB-ADPCM took 2.5 s and 1.6 s. The described and some additional results are listed in Tab. 2.

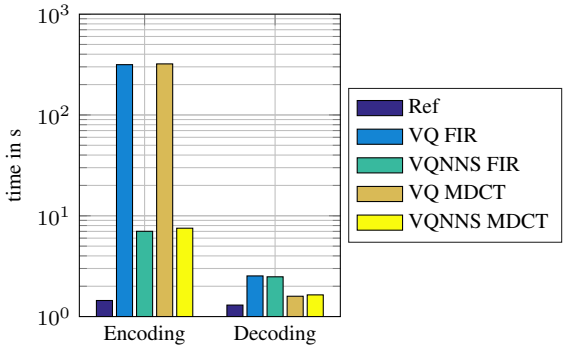


Figure 8: *Averaged encoding and decoding times*

5. CONCLUSION

The goal of this study was to develop a low-delay vector-quantized subband ADPCM codec structure offering a similar or slightly decreased performance at reduced bit rate in comparison to a broadband variant. The codec structure and all its modules were illustrated and optimized in a two-step approach to find meaningful codec parameters, especially for the subband predictors. The resulting initial codec based on a nearly perfectly reconstructing filterbank could reduce the data rate by $1/3$ to 88.2 kbit/s for the cost of 1.1 ms delay and an average degradation of 0.28 ODG score in comparison to the broadband ADPCM codec without noise shaping from [17]. Significant performance gains are expected by further enhancements and extensions of the proposed codec.

For example, different filterbank designs can be considered and evaluated. Since transient signals tend to suffer from the proposed codec structure a transient detection could better control the adaption of the subband predictors. So far only a random codebook combined with nearest neighbor search was implemented but much advanced codebook design algorithms, like pyramid vector quantization [20], can be utilized. Considerable improvement can also be expected when the system is extended with pre- and post-filters to perform noise shaping.

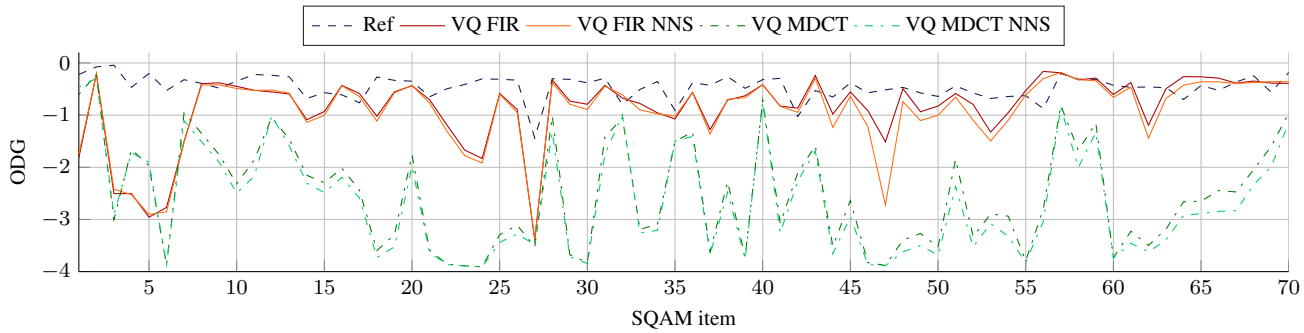


Figure 7: *ODG scores for SQAM items*

Table 2: *Overview of codec evaluation for different configurations for the relevant SQAM items (8-70)*

Configuration	Codebook search	Bit rate in kbit/s	Delay in ms	Mean ODG	Std. Dev. ODG	× realtime
Broadband ADPCM	None	132.3	0	−0.482	0.219	34.75
VQSB-ADPCM FIR	Linear	88.2	1.13	−0.728	0.497	0.159
VQSB-ADPCM FIR	NNS	88.2	1.13	−0.816	0.558	7.16
VQSB-ADPCM MDCT	Linear	88.2	0.18	−2.594	0.971	0.156
VQSB-ADPCM MDCT	NNS	88.2	0.18	−2.763	0.947	6.66

6. REFERENCES

[1] A. Carôt and C. Werner, "Network music performance-problems, approaches and perspectives," in *Proceedings of the Music in the Global Village*, Budapest, Hungary, 2007.

[2] A. Renaud, A. Carôt, and P. Rebelo, "Networked music performance: State of the art," in *Proceedings of the AES 30th International Conference*, Saariselkä, Finland, 2007.

[3] J. Hirschfeld, J. Klier, U. Kraemer, G. Schuller, and S. Wabnik, "Ultra low delay audio coding with constant bit rate," in *AES Convention 117*, San Francisco, USA, Oct 2004.

[4] J. Hirschfeld, U. Kraemer, G. Schuller, and S. Wabnik, "Reduced bit rate ultra low delay audio coding," in *AES Convention 120*, Paris, France, May 2006.

[5] J.M. Valin, G. Maxwell, T. Terriberry, and K. Vos, "High-Quality, Low-Delay Music Coding in the Opus Codec," in *AES Convention 135*, New York, USA, 2013.

[6] J.M. Valin, T. Terriberry, C. Montgomery, and G. Maxwell, "A High-Quality Speech and Audio Codec With Less Than 10-ms Delay," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 1, Jan. 2010.

[7] M. Holters, O. Pabst, and U. Zölzer, "ADPCM with Adaptive Pre- and Post-Filtering for Delay-Free Audio Coding," in *Proc. of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2007)*, Honolulu, USA, April 2007.

[8] M. Holters and U. Zölzer, "Delay-free lossy audio coding using shelving pre- and post-filters," in *Proc. of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2008)*, Las Vegas, USA, March 2008.

[9] M. Holters, C.R. Helmrich, and U. Zölzer, "Delay-Free Audio Coding Based on ADPCM and Error Feedback," in *Proc. of the 11th Conference on Digital Audio Effects (DAFx-08)*, Espoo, Finland, 2008.

[10] D. Martinez, M. Rosa, F. López, N. Ruiz, and J. Curpián, "Sub-band/adpcm audio coder using adaptive vector quantization," in *4th WSES/IEEE World Multiconference on Circuits, Systems, Communications and Computers*, Vouliagmeni, Greece, July 2000.

[11] G. Simkus, M. Holters, and U. Zölzer, "Error resilience enhancement for a robust adpcm audio coding scheme," in *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*, May 2014.

[12] F. Keiler, *Beiträge zur Audiocodierung mit kurzer Latenzzeit*, Cuvillier, 2006.

[13] Henrique S. Malvar, "Modulated qmf filter banks with perfect reconstruction," *Electronics Letters*, June 1990.

[14] Lloyd J. Griffiths, "A continuously-adaptive filter implemented as a lattice structure," in *Proc. of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '77)*, Hartford, USA, May 1977.

[15] D. Cohn and J. Melsa, "The relationship between an adaptive quantizer and a variance estimator (corresp.)," *IEEE Transactions on Information Theory*, vol. 21, Nov 1975.

[16] European Broadcast Union, "EBU Tech. 3253-E: Sound quality assessment material," April 1988.

[17] M. Holters and U. Zölzer, "Automatic parameter optimization for a perceptual audio codec," in *Proc. of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2009)*, Taipei, Taiwan, April 2009.

[18] International Telecommunication Union, "ITU Recommendation ITU-R BS.1387: Method for objective measurements of perceived audio quality," November 2001.

[19] M. Fink, M. Holters, and U. Zölzer, "Comparison of Various Predictors for Audio Extrapolation," *Proc. of the 16th Int. Conference on Digital Audio Effects (DAFx-13)*, 2013.

[20] T. Fischer, "A pyramid vector quantizer," *IEEE Trans. Inf. Theor.*, vol. 32, no. 4, Sept. 2006.