LEARNING TO DEQUANTIZE SPEECH SIGNALS BY PRIMAL-DUAL NETWORKS: AN APPROACH FOR ACOUSTIC SENSOR NETWORKS

Christoph Brauer[⋆] *Ziyue Zhao*[†] *Dirk Lorenz*[⋆] *Tim Fingscheidt*[†]

[⋆] Institute of Analysis and Algebra
[†] Institute for Communications Technology
Technische Universität Braunschweig, Germany
{ch.brauer, ziyue.zhao, d.lorenz, t.fingscheidt}@tu-bs.de

ABSTRACT

We introduce a method to improve the quality of simple scalar quantization in the context of acoustic sensor networks by combining ideas from sparse reconstruction, artificial neural networks and weighting filters. We start from the observation that optimization methods based on sparse reconstruction resemble the structure of a neural network. Hence, building upon a successful enhancement method, we unroll the algorithms and use this to build a neural network which we train to obtain enhanced decoding. In addition, the weighting filter from code-excited linear predictive (CELP) speech coding is integrated into the loss function of the neural network, achieving perceptually improved reconstructed speech. Our experiments show that our proposed trained methods allow for better speech reconstruction than the reference optimization methods.

Index Terms— Speech coding, quantization, machine learning, artificial neural networks

1. INTRODUCTION

The process of representing analog speech signals in the digital domain (e.g., for storage or transmission) is called speech quantization or coding. One aims at efficient, yet high quality coding to save either storage capacity or bitrate, while keeping high speech quality. Hence, algorithms for speech coding thrive to balance computational complexity for encoding and decoding, bitrate, algorithmic latency, and speech quality.

In acoustic sensor networks [1–5], particularly the transmitters must operate with very low computational complexity due to battery lifetime constraints, while some central decoders can consume much higher computational complexity. Hence, this paper aims at improving coding methods that feature low computational complexity at the sender by using simple scalar quantizers, but still use not too heavy computations on the reconstruction side. There are some previous works for the same task: For correlated source signals such as speech, either a time-variant codebook at the receiver [6, 7], a shallow neural network after the decoder [8], or a postprocessor after decoding [9–12] can exploit residual correlations to improve the reconstruction of the quantized signal on the decoder side.

Different from the above methods, our approach builds on a combination of techniques from compressed sensing and convex optimization [5], improving these techniques by unrolling the respective algorithm [13] and using the algorithmic architecture as the basis for a neural network as proposed in [14]. Our approach works on short frames of speech signals which enables realtime capability of the method. Moreover, we use a tailored objective function for the training that builds on the weighting filter from speech codecs based on code-excited linear predictive (CELP) coding, e.g., adaptive multi-rate (AMR) [15] and wideband AMR (AMR-WB) [16]. The weighting filter is originally designed to shape the spectrum of the coding error to follow the speech spectral envelope at some level below, in order to perceptually mask the error [17]. In this work, the reconstruction error between the dequantized speech and the original speech, shaped by the same time-variant weighting filter, is then minimized during the training of the neural network.

The paper is structured as follows: In Section 2, we briefly review the convex optimization problem that was used in [5] for speech dequantization, and propose to unroll the associated iterative optimization algorithm in terms of neural networks in Section 3. In Section 4, we propose a perceptual loss function applying the weighting filter from CELP speech coding for training of the networks. Section 5 presents the simulation setup, the evaluation results, and the discussion. Finally, some conclusions are drawn in Section 6.

2. DEQUANTIZATION BY CONVEX OPTIMIZATION

It has been proposed in [5] to dequantize quantized speech signals by exploiting sparsity of speech in the frequency domain. We use short frames $\mathbf{s}_\ell \in \mathbb{R}^N$ of uniformly quantized speech signals, where N is the frame length and ℓ is the frame index, which is obtained from the respective frame $\tilde{\mathbf{s}}_\ell \in \mathbb{R}^N$ in a speech signal. The authors of [5] proposed to solve the convex optimization problem

$$\mathbf{x}_\ell^\star \in \operatorname*{argmin}_{\mathbf{x}\in\mathbb{R}^N} \|\mathbf{x}\|_1 \quad \text{s.t.} \quad \|\mathbf{K}^{-1}\mathbf{x} - \mathbf{s}_\ell\|_\infty \leq \tfrac{\Delta}{2} \tag{1}$$

where $\mathbf{K}$ denotes the discrete cosine transform (DCT) matrix of dimension $N \times N$, the vector $\mathbf{x}$ is the signal representation in the DCT domain, and Δ is the length of the quantization intervals. The DCT can be expressed as $\mathbf{x}_\ell = \mathbf{K}\mathbf{s}_\ell$, where the elements of the DCT matrix $\mathbf{K}$ are $K_{ij} = \cos\big(\pi \cdot i(j+0.5)/N\big)$, with $i, j \in \{0, 1, \cdots, N-1\}$. The objective $\|\mathbf{x}\|_1$ encourages reconstructions $\mathbf{K}^{-1}\mathbf{x}$ that have a sparse DCT and the constraint in (1) takes into account that $\tilde{\mathbf{s}}_\ell$ originates in an ℓ_∞-norm ball with radius $\Delta/2$ around the quantized signal. Accordingly, $\mathbf{x}_\ell^\star \in \mathbb{R}^N$ approximates $\tilde{\mathbf{s}}_\ell$ in terms of the columns of $\mathbf{K}^{-1}$. Finally, the reconstructed frame of the speech signal is obtained as $\mathbf{s}_\ell^\star \coloneqq \mathbf{K}^{-1}\mathbf{x}_\ell^\star$. We can also perform the change of variables $\mathbf{e} \coloneqq \mathbf{K}^{-1}\mathbf{x} - \mathbf{s}$ and solve the problem

$$\mathbf{e}_\ell^\star \in \operatorname*{argmin}_{\mathbf{e}\in\mathbb{R}^N} \|\mathbf{K}\mathbf{e} + \mathbf{K}\mathbf{s}_\ell\|_1 \quad \text{s.t.} \; \|\mathbf{e}\|_\infty \leq \tfrac{\Delta}{2} \tag{2}$$

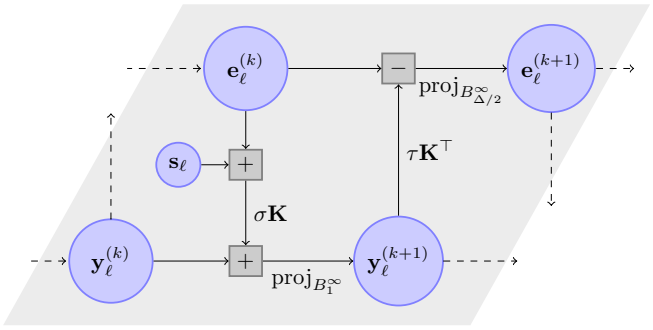

Fig. 1: Primal-dual block without extrapolation.

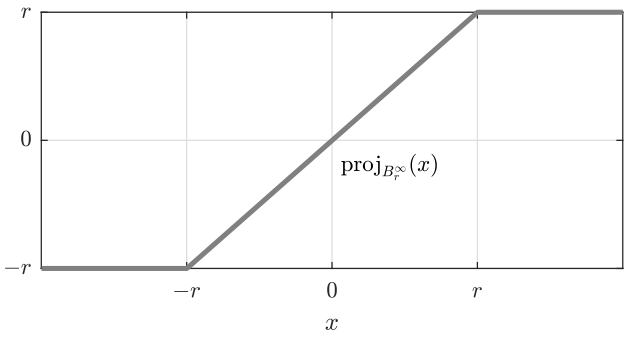

Fig. 2: Projection onto an ℓ_∞-norm ball with radius r.

instead. Applying the primal-dual method of Chambolle and Pock [18] to this problem gives the iteration (iteration index k)

$$\mathbf{y}_\ell^{(k+1)} = \mathrm{proj}_{B^\infty_1}(\mathbf{y}_\ell^{(k)} + \sigma\mathbf{K}(\bar{\mathbf{e}}_\ell^{(k)} + \mathbf{s}_\ell)) \tag{3a}$$

$$\mathbf{e}_\ell^{(k+1)} = \mathrm{proj}_{B^\infty_{\Delta/2}}(\mathbf{e}_\ell^{(k)} - \tau\mathbf{K}^\top\mathbf{y}_\ell^{(k+1)}) \tag{3b}$$

$$\bar{\mathbf{e}}_\ell^{(k+1)} = 2\mathbf{e}_\ell^{(k+1)} - \mathbf{e}_\ell^{(k)}, \tag{3c}$$

where proj_M denotes the orthogonal projection onto M, B^∞_r is the ℓ_∞-norm ball with radius r, and $\tau, \sigma > 0$ are stepsizes. For a one-dimensional input, the utilized projection is

$$\mathrm{proj}_{B^\infty_r}(x) = \max(-r, \min(r, x)) \tag{4}$$

(see Figure 2) and in (3a)–(3b) this kind of projection is applied to each component of the respective vectors. The method is known to converge to a solution of (2) if $\tau\sigma < 1/\|\mathbf{K}\|^2$. As [5] showed, this leads to remarkably good dequantization results for uniformly and non-uniformly quantized signals. In this work we will take this iteration as a starting point for a learned algorithm for dequantization.

3. A NEURAL NETWORK FOR DEQUANTIZATION

As put forward by several works (see, e.g., [13]) one can unroll iterative optimization routines and treat them as neural networks. If we omit the extrapolation step (3c), the respective network is shown in Figure 1. Similar to [14] we propose to use this block as building block for a neural network.

However, different from [14], we keep as much structure of the iteration (3) in the network as possible. More precisely, we just omit the extrapolation step (3c) and treat $\mathbf{K}$, σ and τ as parameters of the network. Especially, we keep the matrix $\mathbf{K}$ tied over all layers and also tied to its transpose in the primal step (3b). Hence, the network works as follows: Given a quantized signal $\mathbf{s}_\ell$ with known length of quantization intervals Δ, the network initializes with $\mathbf{e}_\ell^{(0)} = 0$ and $\mathbf{y}_\ell^{(0)} = 0$ and iterates

$$\mathbf{y}_\ell^{(k+1)} = \mathrm{proj}_{B^\infty_1}(\mathbf{y}_\ell^{(k)} + \sigma\mathbf{K}(\mathbf{e}_\ell^{(k)} + \mathbf{s}_\ell)) \tag{5a}$$

$$\mathbf{e}_\ell^{(k+1)} = \mathrm{proj}_{B^\infty_{\Delta/2}}(\mathbf{e}_\ell^{(k)} - \tau\mathbf{K}^\top\mathbf{y}_\ell^{(k+1)}) \tag{5b}$$

for $k \in \{0, \dots, K-1\}$, where K is the number of primal-dual blocks (5a)–(5b) and the backmost activation $\mathbf{e}_\ell^{(K)}$ is the network output. Due to the above-mentioned change of variables, the reconstructed speech is finally obtained as

$$\hat{\mathbf{s}}_\ell = \mathbf{e}_\ell^{(K)} + \tilde{\mathbf{s}}_\ell\,. \tag{6}$$

Each primal-dual block consists of two consecutive layers with activation functions $\mathrm{proj}_{B^\infty_1}$ and $\mathrm{proj}_{B^\infty_{\Delta/2}}$, respectively. These projection-based activation functions (see Figure 2) are scaled versions of the well-known $\mathrm{hardtanh}$ activation function which was introduced in [19] and can be considered a piecewise linear approximation to the hyperbolic tangent function. Moreover, as pointed out in [14], a primal-dual network consisting of stacked blocks (5a)–(5b) can be understood as two interacting residual networks [20]. However, this holds only if each layer is equipped with its own weights and biases. Here, the parameters of the network are the matrix $\mathbf{K}$ and the stepsizes τ and σ. Note that $\mathbf{K}^\top$ is tied, i.e., we fix that step (5b) uses the transpose of the matrix in step (5a) and also the matrix is the same in each layer. The motivation for the tying of the matrix (and the tying with its transpose) is as follows: Clearly, a network where $\mathbf{K}$ and $\mathbf{K}^\top$ are different and vary with the layers would be more expressive. However, the network would be harder to train (e.g., due to the vanishing gradient problem). More importantly, we already know that the specific choice of the DCT as linear map leads to good dequantization results and hence, we expect that we can still obtain even better results with tied operators. We include the stepsizes σ and τ in the training since we observed in previous experiments in [5] that their choice does influence the performance of the optimization method (3) significantly.

4. DESIGN OF THE LOSS FUNCTION APPLYING THE WEIGHTING FILTER

To obtain a method with realtime capability, we use short frames of speech signals for the training, i.e., we divide the original and the quantized speech signals $\tilde{\mathbf{s}}$ and $\mathbf{s}$ into non-overlapping frames $\tilde{\mathbf{s}}_\ell$ and $\mathbf{s}_\ell$ of length N (and use $N = 320$ at a sampling rate of 16 kHz). These frames of quantized speech signals are fed into our network and the loss function compares the output $\hat{\mathbf{s}}$ of the network with the original speech signal $\tilde{\mathbf{s}}$. A natural loss function would be the mean squares error (MSE) between $\hat{\mathbf{s}}$ and $\tilde{\mathbf{s}}$. It turned out that this loss function leads to an undesirable amount of noise in the reconstruction that lowers the intelligibility significantly.

In order to obtain further improved perceptual quality of the reconstructed speech, a weighted error between the reconstructed speech and the original speech forms the loss function of the neural network. By applying the weighting filter in the loss function, the error between the reconstructed speech and the original speech is trained to track the spectral shape of the *inverse* weighting filter, which actually follows the spectral envelope of the original speech, but is kept at some level below [15]. As a result, this reconstruction error is expected to be less audible due to the auditory masking ef-

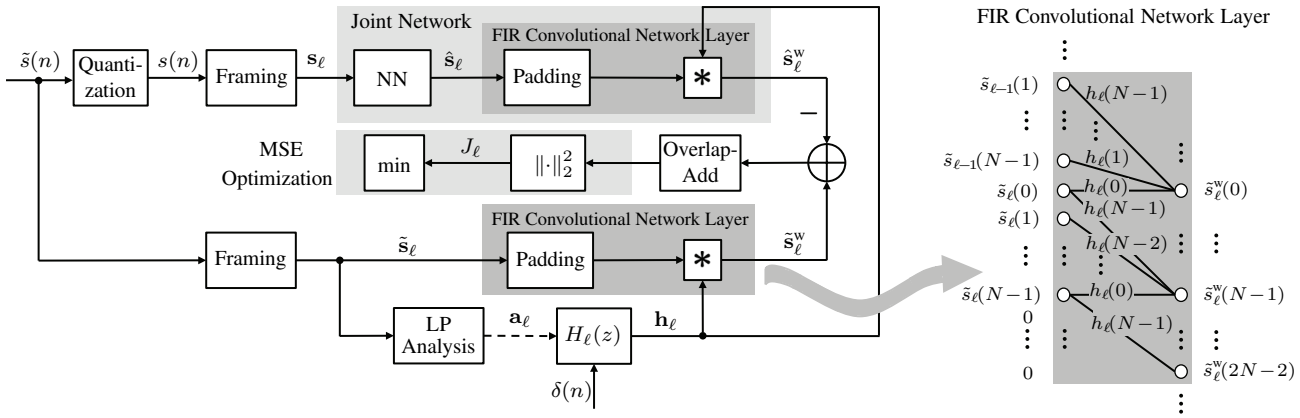

Fig. 3: Framework for **training** with the weighting filter being integrated into the loss function J_ℓ is shown on the left. The detailed structure of the FIR convolutional network layer (in dark grey areas), including padding and convolution ($*$), is shown on the right, which are applied to both the original speech frame $\tilde{\mathbf{s}}_\ell$ and the reconstructed speech frame $\hat{\mathbf{s}}_\ell$ ($\tilde{\mathbf{s}}_\ell$ is taken as an example in this figure). Note that the FIR convolutional network layer is only needed in training, not in test.

fect, which will improve the perceptual quality of the reconstructed speech.

The whole pipeline of the training using this perceptually modified loss function is illustrated in the left part of Figure 3, which will be described in the rest of this section. First, the finite impulse response (FIR) $\mathbf{h}_\ell$ of the weighting filter $H_\ell(z)$ for frame ℓ is obtained as follows: The original speech signal $\tilde{s}(n)$ is assembled from frames $\tilde{\mathbf{s}}_\ell$ without overlap with the frame length being N. After that, a linear prediction (LP) coefficients vector $\mathbf{a}_\ell$ of order (length) 16 is computed from $\tilde{\mathbf{s}}_\ell$ using a rectangular window and the Levinson-Durbin algorithm [21] in LP analysis. Then, the weighting filter $H_\ell(z)$ is obtained by [15]:

$$H_\ell(z) = \frac{A_\ell(z/\gamma_1)}{A_\ell(z/\gamma_2)}, \tag{7}$$

where $A_\ell(z/\gamma) = \sum_{i=1}^{16} a_\ell(i)\gamma^i z^{-i}$, $\gamma_1 = 0.94$ and $\gamma_2 = 0.6$. In order to approximate the FIR $\mathbf{h}_\ell$ of this weighting filter, the delta function[1] $\delta(n)$ is filtered by $H_\ell(z)$, and the first N output samples are regarded as the FIR $\mathbf{h}_\ell$ of this filter.

Subsequently, the weighted speech frames, both the original weighted speech frame $\tilde{\mathbf{s}}_\ell^{\mathrm{w}}$ and the reconstructed weighted speech frame $\hat{\mathbf{s}}_\ell^{\mathrm{w}}$, are obtained by padding and convolution. These two processing steps actually form the FIR convolutional network layer as shown in Figure 3 (dark grey areas). For simplicity, on the right side of Figure 3 we only show how the original weighted speech frame $\tilde{\mathbf{s}}_\ell^{\mathrm{w}}$ is obtained; the reconstructed weighted speech frame $\hat{\mathbf{s}}_\ell^{\mathrm{w}}$ is computed in the same manner. Note that the quantized signal $s(n)$ is assembled from frames $\mathbf{s}_\ell$ and this quantized frame $\mathbf{s}_\ell$ is fed into the neural network (NN) which then provides the reconstructed frame $\hat{\mathbf{s}}_\ell$.

The original speech frame $\tilde{\mathbf{s}}_\ell$ is padded in front with the last $N-1$ samples from the previous frame $\tilde{\mathbf{s}}_{\ell-1}$ and at the end with $N-1$ zeros. This padded vector is then convolved with the FIR $\mathbf{h}_\ell$ from the current frame ℓ, which is denoted as:

$$\tilde{\mathbf{s}}_\ell^{\mathrm{w}}(m) = \sum_{\nu=0}^{N-1} \tilde{s}'(m-\nu)\cdot h_\ell(\nu), \tag{8}$$

[1] $\delta(0)=1$ and $\delta(n)=0$ if $n\neq 0$.

where $m\in\{0,\ldots,2N-2\}$ and $\tilde{s}'(\mu)$ is taken from $\big(\tilde{s}_{\ell-1}(1),\ldots,\tilde{s}_{\ell-1}(N-1),\tilde{s}_\ell(0),\ldots,\tilde{s}_\ell(N-1)\big)$ with indices $\mu\in\{-N+1,\ldots,N-1\}$. This convolution is also illustrated in detail in the right part of Figure 3. Note that the resulting $\tilde{\mathbf{s}}_\ell^{\mathrm{w}}$ has a length of $2N-1$.

Then, the error between the reconstructed weighted speech frame $\hat{\mathbf{s}}_\ell^{\mathrm{w}}$ and the original weighted speech frame $\tilde{\mathbf{s}}_\ell^{\mathrm{w}}$ is overlap-added. Finally, the ℓ_2-norm of the overlap-added error forms the loss function

$$J_\ell(\hat{\mathbf{s}}_\ell,\tilde{\mathbf{s}}_\ell) = \|\mathrm{OLA}\big((\hat{\mathbf{s}}_\ell-\tilde{\mathbf{s}}_\ell)*\mathbf{h}_\ell\big)\|_2^2\,, \tag{9}$$

which is then to be minimized. Note that OLA() stands for the overlap-add operation, where the first $N-1$ samples of frame ℓ are added to the last $N-1$ samples of frame $\ell-1$. The computation of $\|\cdot\|_2^2$ is then only done over the N samples which are readily reconstructed by overlap-add.

5. EXPERIMENTS

We investigate the impact of primal-dual networks on speech using a dataset of 720 sentences from the IEEE corpus [22] consisting of male speech and sampled at 16 kHz. From these signals, 70% (i.e., 504 signals) are used as training set, 15% (i.e., 108 signals, disjoint speakers) are reserved as development set, and another 15% serve as test set (again disjoint speakers) which we use for a comparison with the reconstruction approach proposed in [5] (in Figure 4 referenced as Chambolle-Pock [5]). To that end, we train primal-dual networks with different numbers K of stacked network blocks (5a)–(5b) and compare these to Chambolle-Pock [5] with equations (3a)–(3c) using K iterations. Note that, due to the previously performed change of variables, the reconstructed speech in both cases, is obtained frame-wise as $\hat{\mathbf{s}}_\ell \coloneqq \mathbf{e}_\ell^{(K)} + \mathbf{s}_\ell$.

To make the trained networks principally usable for realtime applications, the original and quantized signals are split into frames using a rectangular window function as described above. We subdivide all 504 signals in the training set this way and end up with $m = 66628$ training examples. In order to learn the network parameters, we experiment with two different loss functions. On the one hand, we use the mean squared error (MSE)

$$J_\ell(\hat{\mathbf{s}}_\ell,\tilde{\mathbf{s}}_\ell) = \|\hat{\mathbf{s}}_\ell-\tilde{\mathbf{s}}_\ell\|_2^2/N\,, \tag{10}$$

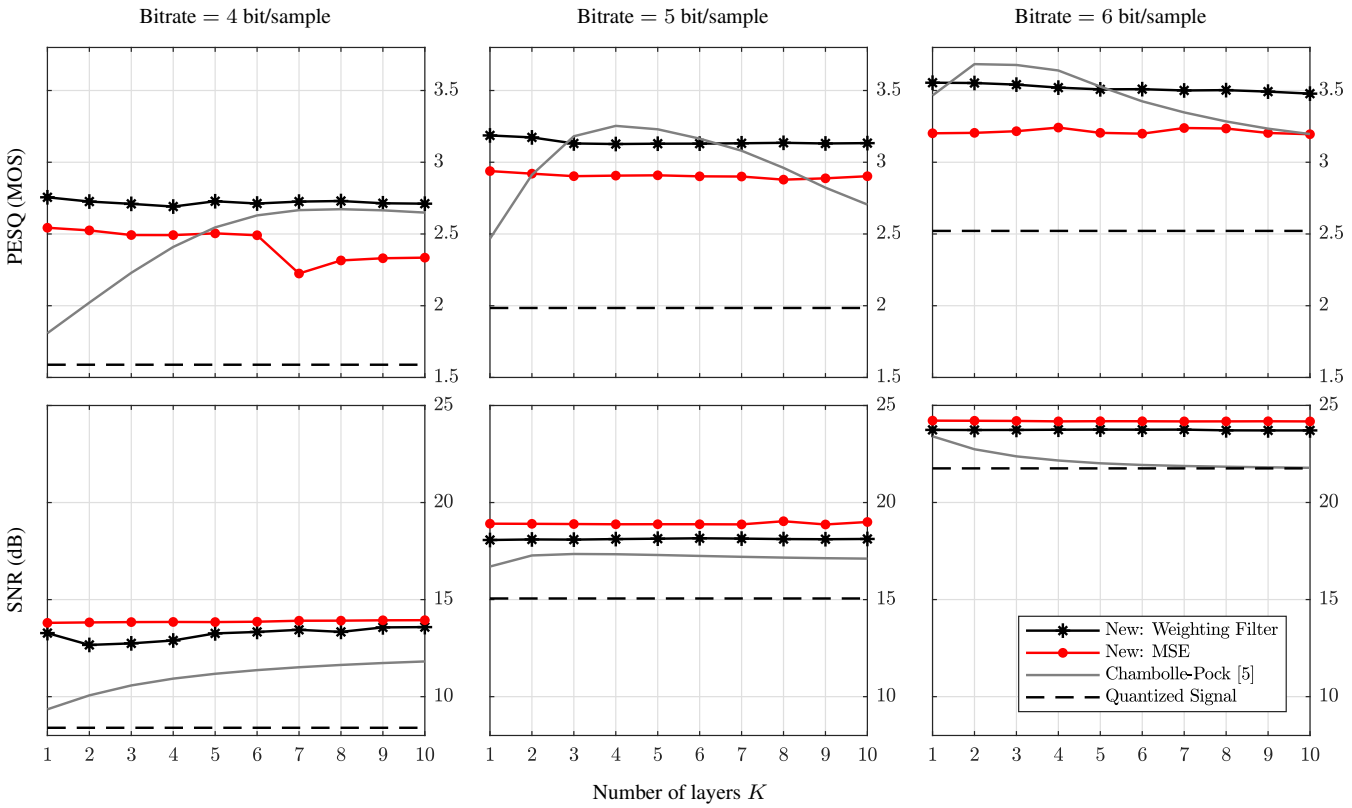

Fig. 4: PESQ and SNR results.

and on the other hand, we use the weighting filter-based loss (9) introduced in Section 4. All experiments were conducted on an NVIDIA GeForce® GTX 1080 Ti GPU using TensorFlow™1.5.0. To minimize the respective overall losses

$$J(\mathbf{K}, \sigma, \tau) = \frac{1}{m} \sum_{\ell=1}^{m} J_\ell(\hat{\mathbf{s}}_\ell, \tilde{\mathbf{s}}_\ell) \tag{11}$$

with respect to $\mathbf{K}$, σ and τ, we perform 3000 epochs of stochastic gradient descent using Adam [23] with learning rate 10^{-4} and all other parameters set to standard values. When using MSE loss, we draw random batches of size 128, whereas one batch comprises all frames associated with a single full signal in case we use the weighting filter-based loss.

Figure 4 illustrates our experimental results. We compare our trained networks against (3a)–(3c) (Chambolle-Pock, [5]) in terms of the Perceptual Evaluation of Speech Quality (PESQ) [24,25] measure (mean opinion score (MOS) listening quality objective (LQO)) and in terms of the signal-to-noise ratio (SNR). To that end, we train 10 different networks for $K \in \{1, \dots, 10\}$ for each of the two utilized loss functions and each of the quantization bitrates 4, 5 and 6 bit/sample. Our results show that the primal-dual networks trained with MSE loss are best in terms of the yielded SNR (which is not surprising due to the close connection between MSE and SNR), closely followed by the networks trained with weighting filter-based loss. Both yield clearly better SNR results than Chambolle-Pock for all considered bitrates. The situation changes when we consider the PESQ measure. Here, the networks trained with weighting filter-based loss are best for small K, especially in case of small bitrates, while Chambolle-Pock can yield better PESQ values for some larger values of K and bitrates 5 and 6 bit/sample. One important observation is that the results for primal-dual networks seem to be almost independent of K, i.e., best results are already obtained when using only one primal-dual block, which is clearly favorable in terms of realtime applicability of the trained networks.

6. CONCLUSION

In this paper, we have proposed to unroll the iterative optimization procedure proposed in [5] for the purpose of speech dequatization in terms of a closely related neural network architecture called primal-dual networks which was proposed in [14]. Moreover, a perceptual loss function for training of the neural network is designed by applying the weighting filter from speech coding. The simulations show that primal-dual networks applied to the task of speech dequantization can outperform the iterative procedure [5] especially for bitrates 4 and 5 bit/sample in terms of the PESQ measure, while they lead to throughout better results in terms of SNR.

7. ACKNOWLEDGMENT

The authors would like to thank Timo Lohrenz and Samy Elshamy for the help concerning the arrangement of neural network training on the GPU cluster.

8. REFERENCES

[1] A. Bertrand and M. Moonen, "Distributed Adaptive Node-Specific Signal Estimation in Fully Connected Sensor Net-

works – Part I : Sequential Node Updating," *IEEE Transactions on Signal Processing*, vol. 58, no. 10, pp. 5277–5291, Oct. 2010.

[2] A. Zahedi, J. Østergaard, S. H. Jensen, S. Bech, and P. Naylor, "Audio Coding in Wireless Acoustic Sensor Networks," *Signal Processing*, vol. 107, pp. 141–152, Feb. 2015.

[3] T. Bäckström, F. Ghido, and J. Fischer, "Blind Recovery of Perceptual Models in Distributed Speech and Audio Coding," in *Proc. of INTERSPEECH*, San Francisco, CA, USA, Sept. 2016, pp. 2483–2487.

[4] Y. Zeng and R. C. Hendriks, "Distributed Delay and Sum Beamformer for Speech Enhancement via Randomized Gossip," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 22, no. 1, pp. 260–273, Jan. 2014.

[5] C. Brauer, T. Gerkmann, and D. Lorenz, "Sparse Reconstruction of Quantized Speech Signals," in *Proc. of ICASSP*, Shanghai, China, March 2016, pp. 5940–5944.

[6] S. Han and T. Fingscheidt, "Improving Scalar Quantization for Correlated Processes Using Adaptive Codebooks Only at the Receiver," in *Proc. of EUSIPCO*, Lisbon, Portugal, Sept. 2014, pp. 386–390.

[7] S. Han and T. Fingscheidt, "An Improved ADPCM Decoder by Adaptively Controlled Quantization Interval Centroids," in *Proc. of EUSIPCO*, Nice, France, Sept. 2015, pp. 335–339.

[8] Z. Zhao, S. Han, and T. Fingscheidt, "Improving Vector Quantization-Based Decoders for Correlated Processes in Error-Free Transmission," in *Proc. of the 12th ITG Conference on Speech Communication*, Paderborn, Germany, Oct. 2016, pp. 70–74.

[9] ITU, *Rec. G.711 Amendment 2: New Appendix III – Audio Quality Enhancement Toolbox*, International Telecommunication Union, Telecommunication Standardization Sector (ITU-T), Nov. 2009.

[10] Z. Zhao, H. Liu, and T. Fingscheidt, "Convolutional Neural Networks to Enhance Coded Speech," *arXiv preprint arXiv:1806.09411*, June 2018.

[11] Z. Zhao, S. Elshamy, H. Liu, and T. Fingscheidt, "A CNN Postprocessor to Enhance Coded Speech," in *Proc. of IWAENC*, Tokyo, Janpan, Sept. 2018, pp. 406–410.

[12] Z. Zhao, H. Liu, and T. Fingscheidt, "Enhancement of G.711-Coded Speech Providing Quality Higher Than Uncoded," in *Proc. of the 13th ITG Conference on Speech Communication*, Oldenburg, Germany, Oct. 2018, pp. 211–215.

[13] K. Gregor and Y. LeCun, "Learning Fast Approximations of Sparse Coding," in *Proc. of International Conference on Machine Learning (ICML)*, Haifa, Israel, June 2010, pp. 399–406.

[14] C. Brauer and D. Lorenz, "Primal-Dual Residual Networks," *arXiv preprint arXiv:1806.05823*, June 2018.

[15] 3GPP, *Mandatory Speech Codec Speech Processing Functions; Adaptive Multi-Rate (AMR) Speech Codec; Transcoding Functions (3GPP TS 26.090, Rel. 14)*, 3GPP; TSG SA, Mar. 2017.

[16] 3GPP, *Speech Codec Speech Processing Functions; Adaptive Multi-Rate-Wideband (AMR-WB) Speech Codec; Transcoding Functions (3GPP TS 26.190, Rel. 14)*, 3GPP; TSG SA, Mar. 2017.

[17] B. Bessette, R. Salami, R. Lefebvre, M. Jelinek, J. Rotola-Pukkila, J. Vainio, H. Mikkola, and K. Järvinen, "The Adaptive Multirate Wideband Speech Codec (AMR-WB)," *IEEE Transactions on Speech and Audio Processing*, vol. 10, no. 8, pp. 620–636, Nov. 2002.

[18] A. Chambolle and T. Pock, "A First-Order Primal-Dual Algorithm for Convex Problems with Applications to Imaging," *Journal of Mathematical Imaging and Vision*, vol. 40, pp. 120–145, May 2011.

[19] Ronan Collobert, *Large Scale Machine Learning*, Ph.D. thesis, Université Paris VI, 2004.

[20] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep Residual Learning for Image Recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.

[21] P. Vary and R. Martin, *Digital Speech Transmission: Enhancement, Coding and Error Concealment*, John Wiley & Sons, 2006.

[22] P. C. Loizou, *Speech Enhancement - Theory and Practice*, CRC Press, Taylor & Francis Group, 2007.

[23] D. P. Kingma and J. L. Ba, "Adam: A Method for Stochastic Optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[24] ITU, *Rec. P.862.2: Wideband Extension to Recommendation P.862 for the Assessment of Wideband Telephone Networks and Speech Codecs*, International Telecommunication Union, Telecommunication Standardization Sector (ITU-T), Nov. 2007.

[25] ITU, *Rec. P.862.2: Corrigendum 1, Wideband Extension to Recommendation P.862 for the Assessment of Wideband Telephone Networks and Speech Codecs*, International Telecommunication Union, Telecommunication Standardization Sector (ITU-T), Oct. 2017.