Bäckström, Tom

End-to-End Optimization of Source Models for Speech and Audio Coding Using a Machine Learning Framework

Published in:
Proceedings of Interspeech

DOI:
10.21437/Interspeech.2019-1284

Published: 01/09/2019

Document Version
Publisher's PDF, also known as Version of record

INTERSPEECH 2019
September 15–19, 2019, Graz, Austria

End-to-end Optimization of Source Models for Speech and Audio Coding Using a Machine Learning Framework

Tom Bäckström

Aalto University, Department of Signal Processing and Acoustics, Finland
first.lastname@aalto.fi

Abstract

Speech coding is the most commonly used application of speech processing. Accumulated layers of improvements have however made codecs so complex that optimization of individual modules becomes increasingly difficult. This work introduces machine learning methodology to speech and audio coding, such that we can optimize quality in terms of overall entropy. We can then use conventional quantization, coding and perceptual models without modification such that the codec adheres to conventional requirements on algorithmic complexity, latency and robustness to packet loss. Experiments demonstrate that end-to-end optimization of quantization accuracy of the spectral envelope can be used for a lossless reduction in bitrate of 0.4 kbits/s.

Index Terms: speech and audio coding, end-to-end optimization, speech source modeling,

1. Introduction

Despite the fact that speech coding is the most commonly used speech processing application – there are an estimated 4.67 billion mobile phone users in 2019[1] – research in coding has dwindled in the academic speech community. A contributing factor to this development is that progress is often published through international standards (e.g. [1,2]), which is a slow process and where it is difficult for individual researchers to participate [3]. A technical reason is that speech codecs have accumulated layers of incremental improvements, which are so densely woven to an interconnected mesh, that introducing new methods often has unexpected side-effects and it is hard to demonstrate the benefit of improvements [4].

The quality of a codec is best measured with subjective evaluations, where human listeners score the output of different codecs [4]. The performance is essentially determined by two models, the perceptual and source models [4, 8]. The perceptual model determines relative quantization accuracy of different variables, while the source model is basically used to losslessly compress the quantized signal. The two models are, to some extent, independent and improvement in one model gives an improvement in overall quality if the other model is fixed. In particular, in this paper we focus on the source model to provide a *reduction in bitrate without changing the quantization* or perceptual model. In terms of entropy coding, such improvements can be characterized by sample entropy, that is, the average log-likelihood of the observation given the source model.

A trend in machine learning has been to optimize systems end-to-end (e.g. [5,6]) such that all components of a system are tuned to optimize a global objective function. If the training uses a sufficiently large database, then this approach provides a level of insurance that complicated interactions between components do not cause unexpected degradations to the system.

[1] https://www.statista.com/statistics/274774/forecast-of-mobile-phone-users-worldwide/

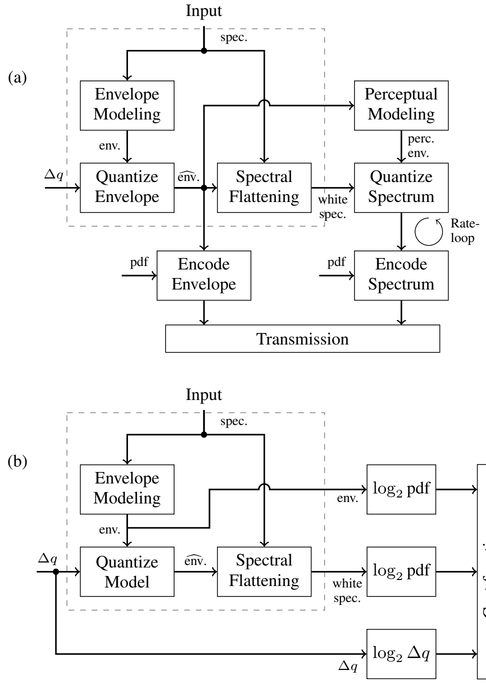

Figure 1: *Flow diagrams of (a) the encoder and (b) off-line training. The dashed square corresponds to the part which are equivalent between both structures. The training process (b) finds those probability distribution functions (pdfs) and that quantization step size Δq which minimizes the cost function (viz. entropy). The encoder (a) takes the pdfs and Δq as inputs from the off-line training (b).*

The main contribution of the current paper is to demonstrate how a conventional speech and audio coding structure, such as [7], can acommodate end-to-end optimization. This enables researchers to evaluate competing methods side by side, without the fear of unintended consequences in other modules. As a first step, here we discuss end-to-end optimization of the source model, encompassing an envelope model and spectral fine structure, and which is used in entropy coding of speech signals. Secondary contributions include improvements to our prior methods in quantization and coding of envelopes [9] and the spectral fine structure [7], which are necessary to make the modules compatible with end-to-end optimization.

The proposed global objective function is the entropy of the source model. Improving the accuracy of the source model is equivalent with improving its entropy, and such improvements can be used for lossless reduction of bitrate. As a consequence, it is here not necessary to consider the effects of quantization or perceptual modeling, nor do we need subjective listening tests

http://dx.doi.org/10.21437/Interspeech.2019-1284

to evaluate quality, but an improvement in entropy suffices.

The design-goal of the current work is to provide a realistic implementation of a speech and audio codec, which can be used on low-power CPUs in both single- and multi-device scenarios [3, 10]. In difference to other machine learning approaches such as WaveNet-based coding [11], we aim to remain within conventional constraints of algorithmic complexity, latency and robustness to packetloss. With the objective of making experiments reproducible and simple, as well as due to space constraints, we have not included more advanced tools, though they are known to improve quality, such as fundamental frequency modeling [4, 12], noise filling [13] and dithering [14, 15].

2. Systems Structure

To conform with the constraints of conventional speech coding environments, our systems model is closely related to conventional speech codecs. Fig. 1a illustrates the flow diagram of the encoder, which takes the spectrum of a single frame as an input. The decoder mirrors the structure of the encoder and is not depicted here. In the encoder, the envelope model characterizes the macro-shape of the power spectrum and the quantized envelope is used to flatten the spectrum. This envelope thus corresponds to the linear predictive (LP) model used in CELP and TCX-type codecs [4]. The quantized envelope is further used to derive a perceptual model, which controls the relative quantization accuracy of the flattened/white spectrum. The absolute quantization accuracy is then optimized in a rate-loop as in [7] to maximize quality within the bit-rate constraint.

The proposed source model is a parametric probability distribution function (pdf), used as input for the entropy coders in envelope and spectrum encoding in Fig. 1a. The parameters of that source model are optimized with an end-to-end training process, whose cost-function is illustrated in Fig. 1b. In coding of the spectrum, quantization accuracy is independent from the probability distribution; by improving the accuracy of the distribution, we obtain a lossless gain in the bitrate, irrespective of the perceptual model. The perceptual model can therefore be omitted from the training process.

The quantized envelope is, however, used for flattening the spectrum, such that the quantization accuracy of the envelope has a direct effect on the distribution of the flattened spectrum. It is therefore a compromise between the bitrate used for coding the envelope versus the spectrum. Consequently, quantization of the envelope must be included in the training phase.

An important difference in the proposed codec with respect to conventional codecs is that the proposed model omits a separate scalar gain term; The overall energy of the signal is coded with the envelope model. The motivation is that inclusion of the gain in the envelope model simplifies the overall structure. This approach also avoids potential overcoding, where the same signal could be encoded in multiple different ways.

The proposed codec uses a fixed quantization step size for the envelope, which requires a variable bit rate (VBR) coder. The remaining bits are used for quantization of the flattened spectrum, where thus a fixed number of bits is available and we need a rate-loop to optimize quality. The quantization step-size is optimized in the off-line training. It should be noted, however, that the conventional rule of thumb in quantization of the envelope is that the mean spectral distortion should be lower than 1 dB [16]. An expected consequence of the current experiments is then to improve the quantization step size based on data driven optimization.

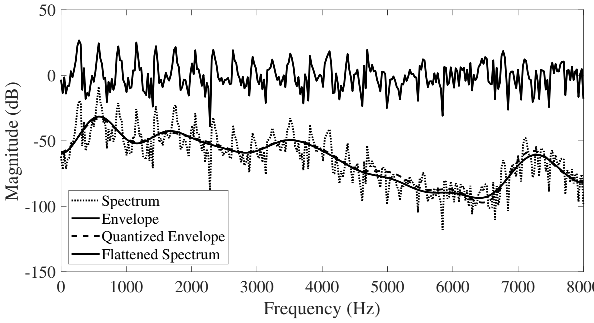

Figure 2: *Example of an input spectrum x, its corresponding envelope model s and the quantized envelope $\hat{s}$ as well as the flattened/whitened spectrum w.*

3. Parametric Signal Model

Let $x \in \mathbb{R}^{N\times 1}$ be the input signal representing the N coefficients of an MDCT-spectrum [4]. To model the envelope, we determine a low-rank cepstral representation $y \in \mathbb{R}^{K}$ as

$$y = D \log\left(|x|\right), \tag{1}$$

where $D \in \mathbb{R}^{K\times N}$ are the K first rows of a discrete cosine transform (DCT) matrix of type II. The kth envelope parameter y_k is quantized as

$$\hat{y}_k = \Delta q \ \mathrm{round}(y_k/\Delta q), \tag{2}$$

where the operator $\mathrm{round}()$ signifies rounding to the nearest integer and Δq correspond to the quantization step size. The output envelope $\hat{s} \in \mathbb{R}^{N\times 1}$ is then $\hat{s} = \exp\left(D^T \hat{y}\right)$. It follows that $\hat{s}$ has the approximate gross shape of the speech spectrum (see Fig. 2). By sample-wise division $w_k = x_k/\hat{s}_k$ we obtained a flattened version of the spectrum $w \in \mathbb{R}^{N\times 1}$, where the envelope shape and the signal energy have been normalized or removed (see Fig. 2).

As a parametric model of the probability distribution, we use mixtures of logistic distribution functions [17], where the cumulative probability $c(\xi)$ and the probability distributions $f(\xi)$ are defined, respectively, as

$$c(\xi;\mu;\sigma) = \frac{1}{1+e^{-\frac{\xi-\mu}{\sigma}}}, \ \text{ and } \ f(\xi;\mu;\sigma) = \frac{\partial c(\xi;\mu;\sigma)}{\partial \xi}. \tag{3}$$

For the envelope parameters y we use a multivariate, logistic mixture model whose probability distribution function is

$$f(y) = \prod_{k=1}^{K} \sum_{m=1}^{M} \gamma_{k,m} f(w_k;\mu_{k,m};\sigma_{k,m}). \tag{4}$$

Similarly, for the flattened spectrum w we use the distribution model

$$f(w) = \prod_{n=1}^{N} \sum_{l=1}^{L} \alpha_{n,l} f(w_n;\beta_{n,l};\tau_{n,l}). \tag{5}$$

Here the mixture weights must be positive and add up to unity, $\gamma_{k,m} \geq 0$, $\alpha_{n,l} \geq 0$, $\sum_{m=1}^{M} \gamma_{k,m} = 1$ and $\sum_{l=1}^{L} \alpha_{n,l} = 1$ and scale-factors must be strictly positive, $\sigma_{m,k} > 0$ and $\tau_{l,k} > 0$. Parameters representing component means, the scalars $\mu_{m,k} \in \mathbb{R}$ and $\beta_{l,k} \in \mathbb{R}$ are constrained only to the field of real values.

4. Training

The parameters of the signal model are optimized with a single criteria; maximum likelihood of observation. In other words, we minimize the entropy, measured in bits, which thus directly corresponds to minimizing the bitrate. In typical coding applications, improving the entropy gives a lossless reduction in bitrate. This is also true for the flattened spectral components in the current case; we do not have to include quantization of components in the cost function, since improving the entropy gives a lossless reduction in bitrate whatever the quantization is.

For the envelope parameters, the situation is however more complicated. The quantized envelope parameters are used as pre-conditioning (flattening) for the spectrum (see Fig. 1), such that efficiency of the pre-conditioning influences the entropy of the spectral components; Bits used for encoding the envelope model reduce the number of bits required for encoding the spectrum. To strike a compromise between the cost of encoding the two, we must include the quantization accuracy of the envelope parameters in our cost-function.

The $\log_2$-likelihood of the envelope parameters and flattened spectral components are $\log_2 f(y)$ and $\log_2 f(w)$, respectively. For the quantization accuracy Δq, we are not interested in the absolute bitrate required, but we need only the sensitivity of the entropy with respect to Δq. We can readily see that the sensitivity of the bitrate to changes of Δq is $K \log_2 \Delta q$. The cost-function for the training can then be defined as

$$\begin{aligned} &\mathrm{cost}(x; \mu_{m,k}, \sigma_{m,k}, \gamma_m, \beta_{l,k}, \tau_{l,k}, \alpha_{l,k}, \Delta q) \\ &\qquad = -\left[\log_2 f(y) + \log_2 f(w) + K \log_2 \Delta q\right]. \end{aligned} \tag{6}$$

Reductions in this cost-function thus gives a lossless improvement in overall bitrate.

5. Codec

The coder consists of two parts, a variable bit-rate coder for the envelope parameters and a fixed bit-rate coder for the spectral coefficients, where the latter uses all the bits remaining after encoding the envelope. Specifically, a parameter $\xi \in \mathbb{R}$ which follows a logistic mixture model, has the cumulative distribution function

$$c(\xi) = \sum_{j=1}^{J} \rho_j c(\xi; \delta_j; \lambda_j). \tag{7}$$

If ξ is then quantized to a bin $\xi \in [q_t,\ q_{t+1}]$, then the probability of that bin is

$$P\left[\xi \in [q_t,\ q_{t+1}]\right] = c(q_{t+1}) - c(q_t), \tag{8}$$

which is exactly what is needed to entropy code the quantized value with an arithmetic coder [4, 18]. We encode all envelope parameters with the fixed quantization bin size Δq, such that no further steps are necessary.

The spectral coefficients, in turn, are then quantized such that the accuracy follows the perceptual model and encoded with a fixed-rate codec achieved by a rate-loop, following [7]. Here we use uniform quantization to keep the system simple and allow straightforward comparison to prior methods, even if it is clear that dithered quantization for the low-bitrate parts would improve quality [14].

The first step is to estimate the entropy of the spectral coefficients which follow Eq. 5. A simple analytic form for the entropy of a logistic mixture model is not available, whereby we estimate the entropy as follows. We calculate the probabilities p_k of evenly spaced quantization bins with a step size $1/Q$ using $Q = 256$. We can then calculate the entropy H_{256} of the histogram over the training set as

$$H_Q = \sum_{q=1}^{Q} -p_k \log_2 p_k. \tag{9}$$

We can then approximate the entropy of an arbitrary quantization accuracy Q by

$$H_Q = H_1 + \log_2 Q = H_{256} - 8 + \log_2 Q. \tag{10}$$

In other words, with the measured H_{256} we can determine H_1 and thus find the entropy H_Q of an arbitrary quantization accuracy Q. Conversely, we will determine the entropy H_1 for each spectral component. These entropy values then tells us the relative number of bits $H_r(f)$ required to encode each component f to achieve uniform accuracy.

The perceptual masking model then provides an envelope shape $W(f)$, which corresponds to the relative magnitude of quantization errors that gives a uniform perceptual degradation [4, 19]. The number of bits required to quantize a sample with error $W(f)$ is relative to $\log_2 W(f)$. However, since we know that spectral coefficients need $H_r(f)$ bits to achieve uniform accuracy, we add this as a bias correction and define the perceptually weighted bit-consumption for each spectral component as

$$B(f) := \log_2 W(f) + H_r(f). \tag{11}$$

Note that the definition of $B(f)$ did not take into account the target bitrate and indeed, the perceptual envelope offers only a relative envelope shape, but not an absolute level for the envelopes. In other words, we must further correct the envelope to match the target bit-rate. We choose a correction term η and define

$$B(f, \eta) := \log_2 W(f) + H_r(f) + \eta. \tag{12}$$

Our objective is to determine η such that we reach a target bitrate $B_{\text{target}} = \sum_{f=0}^{F-1} B(f, \eta)$. Clearly this is achieve with $\eta = \frac{1}{F} \sum_{f=0}^{F-1} B(f, 0)$. However, this can result in a non-realizable negative bitrate $B(f, \eta) < 0$. We therefore set any negative bitrates to zero and repeat the process, determine η for the remaining coefficients, until all $B(f, \eta)$ are non-negative.

This procedure gives us a target bitrate for each spectral coefficient. Then we must still quantize each coefficient with the corresponding accuracy. From Eq. 10 we find that the desired quantization step size $1/Q(f)$ is found by $Q(f) = 2^{B(f,\eta)}$. This accuracy gives, on average, the target bitrate, but to match the bitrate of a sample with the target, we therefore need a rate-loop, where the input spectrum is scaled such that we reach the highest accuracy with the given bitrate; see [1, 7] for details.

6. Experiments

To evaluate the performance of the system, we implemented it using a sampling rate of 16 kHz, window length of 30 ms, window step of 20 ms, and a half-sine window with a flat top of 10 ms [4]. Each window is transformed to the frequency domain using the MDCT [4, 19]. Envelopes, including signal gain, are modeled in the cepstral domain with 20 coefficients. We chose to use logistic mixture models for the envelope and spectrum with, respectively, $M = 5$ and $L = 3$ mixture components.

The perceptual model in the codec is based on that of the EVS codec [1], which is based on a linear predictive model of the signal. Since the current codec does not have a predictive model, we estimated the signal autocorrelation from the power

Table 1: *Differential entropy of proposed and reference models.*

Envelope	Proposed	GMM
Mixture order $M = 5$	75.1 bits	72.0 bits

Flattened spectrum	Proposed	Gaussian	Laplacian
Optimal Δq	990 bits	1024 bits	1002 bits
Conventional Δq	979 bits	1006 bits	993 bits

Table 2: *Statistics of the bitrate of the quantized envelope.*

Bitrate	Mean	Standard deviation
Optimal Δq	44.0 bits	16.2 bits
Conventional Δq	63.0 bits	15.9 bits

spectrum of the quantized envelope using an inverse DFT and then used the Levinson-Durbin algorithm to obtain the coefficients of the corresponding predictive model [4]. Though it is unlikely that such a computationally complex method would be used in a real codec, it does provide a reliable reference quality for experiments.

Parameters were trained and tested over the corresponding sets of the TIMIT corpus [20]. The system was trained with batches of 1000 frames and the ordering of frames was randomized before each epoch. As a safeguard for saturation effects, quantization of the envelope was in training replaced by addition of uniformly distributed noise on the same range as the corresponding quantization bin. The cost function of minimum entropy was optimized with the Adam-algorithm, in the Tensorflow-environment on a desktop computer. Informal observations show that the computational complexity of training was reasonable and the model converged with less than 10 epochs and within 15 min of computations.

The differential entropy of the trained models, evaluated over the test set, are listed in Table 1. Observe that the absolute values of differential entropies are generally not meaningful and that we should compare only the differences in entropy. We find that, for the cepstral envelope parameters, the proposed logistic mixture model requires 3.1 bits/frame more than a diagonal Gaussian mixture model with same number of parameters, which corresponds to an rather negligible increase in bitrate of 0.155 kbits/s. The main advantage of the proposed method is then that the cumulative distribution is simple to calculate with Eq. 7, whereas the Gaussian model would require computations of the error function, which can not be expressed in terms of elementary functions and approximations are complicated.

Table 1 also shows the mean differential entropy of the flattened spectral components per frame. It shows that, when using the optimal quantization step Δq, the proposed mixture model gives an advantage of 12 bits/frame and 34 bits/frame over Laplacian and Gaussian distributions, respectively, which correspond to 0.6 kbits/s and 1.7 kbits/s. When the quantization step is manually tuned such that the average distortion is below 1 dB, to follow the conventional rule-of-thumb [16], we see that the differential entropy of both the proposed model and the Laplacian model are reduced with approximately 10 bits and the Gaussian by 18 bits.

Table 2 lists the bitrate statistics of the envelope with the optimal and conventional quantization step sizes Δq. Obviously, since the optimal quantization accuracy is much lower, also the mean bitrate is reduced by 19 bits. The same is reflected in the mean spectral distortion of the quantized envelopes, listed in Table 3. A reduction of the accuracy increases the spectral distortion of the envelope as expected. Since we save approximately 19 bits in encoding the envelope, it offsets the 11 bits increase in differential entropy for the flattened spectrum. In conclusion, optimization of Δq thus reduces the bitrate by a total of 8 bits/frame or 0.4 kbits/s. However, the advantage is available only when using the logistic mixture model for the flattened spectrum and disappears if using the simpler Laplacian model.

Table 3: *Spectral distortion (SD) of the quantized envelope.*

	Mean SD	in 2–4 dB	above 4 dB
Optimal Δq	2.0 dB	47 %	0 %
Conventional Δq	1.0 dB	0 %	0 %

Table 4: *Mean perceptual signal to noise ratio.*

Bitrate	13.2 kbits/s	16 kbits/s	24 kbits/s
Optimal Δq	2.44 dB	3.05 dB	5.19 dB
Conventional Δq	2.30 dB	2.91 dB	4.99 dB

To estimate the perceptual effect of proposed model, we further measured the perceptual signal to noise ratio of the output signal for typical bitrates (see Table 4). The perceptual model is the same as that used for perceptual quantization. This provides an objective measure which approximates subjective quality. We observe that the optimal quantization step size provide an improvement in quality of approximately 0.2 dB. According to our experience, this difference in quality is very close to the just noticeable difference for experienced listeners. It is therefore not worthwhile to perform a subjective listening test, since it is unlikely that we would get a statistically significant difference between methods.

7. Conclusions

Tuning speech and audio codecs has become increasingly difficult in tandem with their complexity. The current work proposes an end-to-end approach for optimizing the source model in such codecs, such that the entropy of the whole codec can be minimized. As a demonstration, we focus here on the quantization accuracy of the spectral envelope, which is in conventional codecs chosen using a rule-of-thumb. Our experiments show that there is more freedom in the choice in that quantization accuracy than previously thought and a reduction of 0.4 kbits/s can be achieved just by this optimization. Conversely, we also find that quantization within the network corresponds to addition of noise, which is in the machine learning community known as regularization and here the amount of regularization is present in the cost function. Classical tools of machine learning thus have well-warranted counterparts in coding.

An important benefit of the proposed methodology is thus that now speech and audio coding is framed as a machine learning problem. Connecting these two fields opens plenty of opportunities for improvement. For example, obvious next steps include at least introduction of conventional tools such as fundamental frequency models, bandwidth extension and either noise filling or dithering [4, 14, 15]. Conversely, we can introduce methods from machine learning, such as adding new layers to the network to improve latent representations.

8. Acknowledgments

This work was supported by the Academy of Finland project No 312490.

9. References

[1] *TS 26.445, EVS Codec Detailed Algorithmic Description; 3GPP Technical Specification (Release 12)*, 3GPP, 2014.

[2] ISO/IEC 23003–3:2012, "MPEG-D (MPEG audio technologies), Part 3: Unified speech and audio coding," 2012.

[3] T. Bäckström, "Speech coding, speech interfaces and IoT – opportunities and challenges," in *52nd Asilomar Conference on Signals, Systems and Computers (invited paper)*, 2018.

[4] ——, *Speech Coding with Code-Excited Linear Prediction*. Springer, 2017.

[5] B. Dhingra, L. Li, X. Li, J. Gao, Y.-N. Chen, F. Ahmed, and L. Deng, "Towards end-to-end reinforcement learning of dialogue agents for information access," in *Proc 55th Annual Meeting of the Association for Computational Linguistics*, vol. 1, 2017, pp. 484–495.

[6] J. D. Williams, K. Asadi, and G. Zweig, "Hybrid code networks: practical and efficient end-to-end dialog control with supervised and reinforcement learning," in *Proc 55th Annual Meeting of the Association for Computational Linguistics*, vol. 1, 2017, pp. 665–677.

[7] T. Bäckström and C. R. Helmrich, "Arithmetic coding of speech and audio spectra using TCX based on linear predictive spectral envelopes," in *Proc. ICASSP*, Apr. 2015, pp. 5127–5131.

[8] S. Disch, S. van de Par, A. Niedermeier, E. Burdiel Pérez, A. Berasategui Ceberio, and B. Edler, "Improved psychoacoustic model for efficient perceptual audio codecs," in *Audio Engineering Society Convention 145*. Audio Engineering Society, 2018.

[9] S. Korse, G. Fuchs, and T. Bäckström, "GMM-based iterative entropy coding for spectral envelopes of speech and audio," in *Proc. ICASSP*, 2018.

[10] T. Bäckström, F. Ghido, and J. Fischer, "Blind recovery of perceptual models in distributed speech and audio coding," in *Proc. Interspeech*, 2016, pp. 2483–2487.

[11] W. B. Kleijn, F. S. Lim, A. Luebs, J. Skoglund, F. Stimberg, Q. Wang, and T. C. Walters, "Wavenet based low rate speech coding," *arXiv preprint arXiv:1712.01120*, 2017.

[12] T. Moriya, Y. Kamamoto, N. Harada, T. Bäckström, C. F. Helmrich, and G. Fuchs, "Harmonic model for MDCT based audio coding with LPC envelope," in *Proc. EUSIPCO*, 2015.

[13] S. Disch, A. Niedermeier, C. R. Helmrich, C. Neukam, K. Schmidt, R. Geiger, J. Lecomte, F. Ghido, F. Nagel, and B. Edler, "Intelligent gap filling in perceptual transform coding of audio," in *Audio Engineering Society Convention 141*. Audio Engineering Society, 2016.

[14] T. Bäckström, J. Fischer, and S. Das, "Dithered quantization for frequency-domain speech and audio coding," in *Proc. Interspeech*, 2018, pp. 3533–3537.

[15] T. Bäckström and J. Fischer, "Fast randomization for distributed low-bitrate coding of speech and audio," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 26, no. 1, January 2018.

[16] K. K. Paliwal and B. S. Atal, "Efficient vector quantization of LPC parameters at 24 bits/frame," *IEEE Trans. Speech Audio Process.*, vol. 1, no. 1, pp. 3–14, 1993.

[17] C. Walck, *Handbook on statistical distributions for experimentalists*. University of Stockholm Internal Report SUF-PFY/96-01, 2007.

[18] J. Rissanen and G. G. Langdon, "Arithmetic coding," *IBM Journal of research and development*, vol. 23, no. 2, pp. 149–162, 1979.

[19] M. Bosi and R. E. Goldberg, *Introduction to Digital Audio Coding and Standards*. Kluwer Academic Publishers, 2003.

[20] J. S. Garofolo, L. D. Consortium *et al.*, *TIMIT: acoustic-phonetic continuous speech corpus*. Linguistic Data Consortium, 1993.