A TWO-STAGE MLP+NLMS LOSSLESS CODER FOR STEREO AUDIO

Emmanuel Ravelli *, *Philippe Gournay and Roch Lefebvre*

Department of Electrical and Computer Engineering
University of Sherbrooke
Sherbrooke (Québec) J1K 2R1 CANADA
Philippe.Gournay@Usherbrooke.ca

ABSTRACT

We propose in this paper a two-stage lossless audio coder for stereo signals. The first stage is based on a hybrid approach that uses either a stereo linear prediction or a multilayer perceptron-based (MLP) nonlinear prediction. The second stage is based on two cascaded Normalized Least Mean Square (NLMS) filters that remove the remaining redundancy in the residuals of the stereo prediction. The obtained compression ratios are equal or superior to the best state-of-the-art coders.

1. INTRODUCTION

State-of-the-art lossless audio coders (e.g. MPEG-4 ALS [1], LPAC [2], Monkeys [3]) use linear prediction to remove redundancy from an audio signal. In earlier coders, linear prediction was applied on mono signals or on each channel of stereo signals. To take advantage of the correlation that exists between the two channels of a stereo signal, a technique called joint-channel linear prediction has been proposed in [4] and then implemented in the LPAC coder [5]. The idea is to perform simultaneously an intra- and interchannel decorrelation. This technique has shown very good results; however, the predictor coefficients can't be quantified using efficient techniques such as line spectral frequencies (LSFs) and it is necessary to code them by scalar quantization with at least 12 bits per coefficient to keep the decoder stable. To get around the transmission of the coefficients, a backward approach was proposed in [6]. In this approach, the coefficients are estimated on the previous frame of the audio signal, instead of on the current frame, but the resulting predictor is applied to the current frame. Since the encoder and the decoder can perform exactly the same operation on the past audio signal, there is no need to quantize and transmit the prediction coefficients as in the forward approach. The results obtained were promising and we follow this approach in our paper.

We first propose an extension of this technique to the nonlinear case using neural networks. This backward nonlinear stereo prediction is the basis of the first stage of our coder. Such types of nonlinear predictions have been studied for speech coding purposes in the papers of Faundez et al. (see for example [7, 8]).

As the residuals of the nonlinear prediction are not fully decorrelated, we propose a second stage based on two cascaded NLMS filters for each residual. Cascaded adaptive linear prediction for lossless audio coding was first proposed in [9] and [10]. In these papers, the prediction was performed directly on an input mono signal. We use a similar technique to remove the redundancy that remains in the residuals of the joint-channel prediction. This second stage improves greatly the decorrelation of the stereo signal.

The remainder of this paper is structured as follows. Section 2 presents the first stage of our coder. Section 3 presents the second stage. Results are given in Section 4 and conclusions in Section 5.

2. FIRST STAGE: HYBRID LINEAR/MLP STEREO PREDICTION

The first stage is based on a hybrid approach using, for each frame and for each channel, either a joint-channel linear prediction or a joint-channel nonlinear prediction; the model which maximizes the prediction gain is chosen. A backward analysis is used: the predictor coefficients are estimated from the past decoded signal, which is available at both the encoder and the decoder. The backward analysis has the advantage to avoid the quantization and the transmission of the prediction coefficients as in the forward approach. We first outline the nonlinear prediction we use which is based on neural networks. Then the hybrid approach is introduced.

2.1. The neural network structure

The choice of the neural network structure is very important because it determines the performance of the predictor. Indeed, in a backward configuration, the network is trained on a frame and tested on the next frame. The network does not have to be specific only for the data used for the training, it has to be able to generalize. If the network models the frame

*E. Ravelli is now with the Laboratoire dAcoustique Musicale, Universite Pierre et Marie Curie, Paris, France. (e-mail: ravelli@lam.jussieu.fr).

used for the training perfectly but is not able to model the test frame, the network is said to be overtrained [8]. To limit overtraining, a very simple but efficient structure is used: a Multi-Layer Perceptron (MLP) network with one hidden layer comprising one neuron. Its input-output function is:

$$\hat{x}[n] = w_{2,1} f\left(\sum_{k=1}^{P} w_{1,k} x[n-k]\right) + b \qquad (1)$$

with $x[n-k]$ the input samples, $\hat{x}[n]$ the predicted sample, P the prediction order, f the transfer function of the hidden neuron (sigmoid), $w_{1,k}$ the weights of the first layer and $w_{2,1}$ and b the weight and the bias of the second layer.

2.2. Extension to the stereo case

By analogy with the joint-channel linear prediction, we propose a joint-channel nonlinear prediction using two neural networks, one for each channel. The input-output function of the left-channel network is:

$$\hat{x}_1[n] = w_{2,1} f(y) + b \qquad (2)$$

$$y = \sum_{k=1}^{P_1} w_{1,k} x_1[n-k] + \sum_{k=1}^{P_2} w_{1,k+P_1} x_2[n-k] \qquad (3)$$

with $x_1(n)$ the left-channel signal, $x_2(n)$ the right-channel signal, $\widehat{x}_1(n)$ the predicted left channel signal, P_1 the autopredictor order and P_2 the crosspredictor order. For the right channel, as the left and right channels are interleaved, the new sample $x_1(n)$ can be used to predict $x_2(n)$ [5]. The sum corresponding to the crosspredictor consequently starts at index 0 rather than at index 1 in the following equation. This technique greatly improves the prediction gain of the right channel if the channels are highly correlated. The input-output function of the right-channel network is:

$$\hat{x}_2[n] = w'_{2,1} f(y) + b' \qquad (4)$$

$$y = \sum_{k=1}^{P_1} w'_{1,k} x_2[n-k] + \sum_{k=0}^{P_2} w'_{1,k+P_1+1} x_1[n-k] \qquad (5)$$

2.3. Initialization and training of the neural networks

To better adapt the model to the signal, a "subframe" analysis is performed. The predictor parameters are updated four times per frame. Consequently, if the length of a frame is N, the parameters are kept constant on a "subframe" of length $N/4$ (Fig. 1).

To initialize the weights and bias of the neural network, we use the neural network parameters computed from the previous frame. As the successive frames are quite similar, this method greatly increases the training of the network compared to a random initialization.

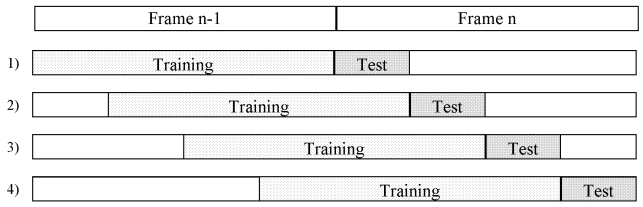

Fig. 1. The principle of subframe analysis for the MLP predictor

To train the neural networks, the Levenberg-Marquardt algorithm is used. This algorithm is very fast. But its computational cost increases greatly with the size of the network, and it is usually used only for small networks like the one in this paper. To limit overtraining, the number of epochs (or iterations) of the training algorithm is set to only 10. This also limits the complexity.

2.4. Hybrid approach

In some frames, linear prediction gives a better prediction gain than nonlinear prediction (neural networks). Thus, we use a hybrid approach as in [7]: for each frame and for each channel, two analyses are performed (one linear and one nonlinear), the prediction gains obtained with each analysis are then calculated, and finally, for each channel, the best predictor is chosen. The linear prediction used is the same as in [6]: the analysis is performed on an extended frame weighted with an asymmetric window. The Cholesky decomposition is used to invert the matrix. In order for the decoder to know which prediction was used for each channel, two bits per frame are sent to the decoder.

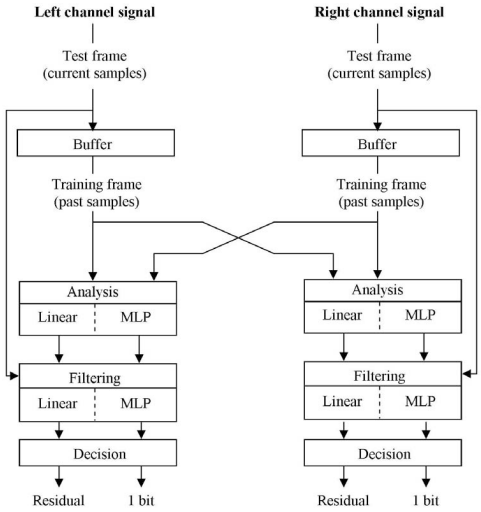

Fig. 2. First stage: hybrid Linear/MLP stereo prediction

3. SECOND STAGE: CASCADED NLMS FILTERING

The second stage is based on two cascaded NLMS filters for each channel. These filters remove the redundancy that remains in the residuals of the joint-channel prediction. An NLMS filter is the normalized version of the standard LMS filter. This algorithm is summarized in Equations (6) to (9):

1. Initialization

$$\mathbf{h}_0 = 0 \tag{6}$$

2. Loop

$$\widehat{x}(n) = \mathbf{h}_n^T \mathbf{x}_n \tag{7}$$

$$e(n) = x(n) - \widehat{x}(n) \tag{8}$$

$$\mathbf{h}_{n+1} = \mathbf{h}_n + \mu \frac{\mathbf{x}_n}{\delta + \|\mathbf{x}_n\|} e(n) \tag{9}$$

with $x(n)$ the input signal (first stage residual in the left or right channel), $\mathbf{x}_n = [x(n-1), ..., x(n-P)]^T$ the vector corresponding to P past values of the input signal, P the order of the filter, $\mathbf{h}_n = [h(1), ..., h(P)]^T$ the filter coefficients vector, $e(n)$ the prediction error and δ a small constant that avoids a division by 0. Constant μ is a parameter we fix experimentally in order to maximize the average compression ratio over several test audio files. The best results are obtained using two cascaded NLMS filters for each prediction error, the first one having an order of 200 with $\mu = 0.07$, the second having an order of 10 with $\mu = 0.03$. The overall scheme of the two-stage coder is shown in Fig. 3. The Rice code [11] is used to code the samples of the prediction residuals.

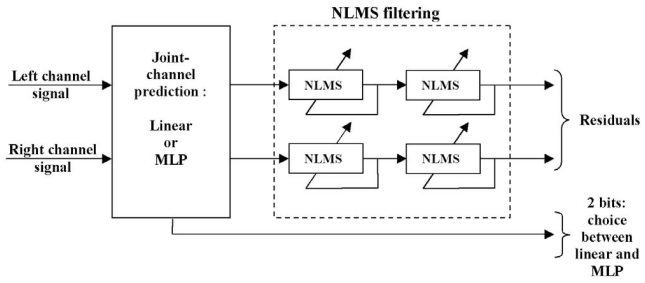

Fig. 3. The proposed two-stage lossless audio coder

4. EXPERIMENTAL RESULTS

4.1. Prediction gains

To limit the complexity of the system, we found that the following values are a good compromise: an autoprediction order of $P_1 = 20$, an interprediction order of $P_2 = 10$. The frame length is 20 ms (N=960 at 48kHz). The analysis is performed four times a frame, leading to a 5 ms subframe analysis.

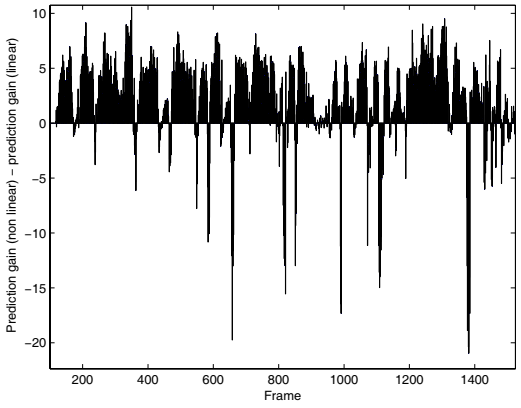

Fig. 4. Comparison of the linear and non-linear prediction gains

Fig. 4 compares, for each frame of the left channel of a speech signal (48kHz), the gains obtained with the joint-channel MLP predictor described in the previous section to the gains obtained with the joint-channel linear predictor of [6]. We notice that the gain compared to the linear prediction is not always positive. For several frames, the linear prediction gives better prediction gains than the nonlinear prediction. For a speech signal, it generally corresponds to the unvoiced parts of the signal. This illustrates well why we use a hybrid approach (mix of linear and non-linear prediction).

Average prediction gains obtained with the linear and the hybrid approaches are shown in Table 1. The signals used for these experiments are taken from an MPEG test base, and all are sampled at 48 kHz. These results reveal the superiority of the hybrid approach, which clearly outperforms the linear predictor.

Speech	**Linear**	**Hybrid**
Left channel	22.19 dB	+2.43 dB
Right channel	39.09 dB	+8.03 dB
Pop music		
Left channel	22.45 dB	+3.81 dB
Right channel	19.17 dB	+3.38 dB
Classical music		
Left channel	29.51 dB	+6.59 dB
Right channel	30.58 dB	+7.06 dB

Table 1. Comparison of average prediction gains

The additional average prediction gains obtained with the cascaded NLMS filters are shown in Table 2. The signals used are the same as for the previous experiment. These results show well that the residuals of the joint-channel prediction are not fully decorrelated and the cascaded NLMS filtering

can greatly improve the performance of the coder.

Speech	**NLMS**
Left channel	1.54 dB
Right channel	2.51 dB
Pop music	
Left channel	3.32 dB
Right channel	3.13 dB
Classical music	
Left channel	5.12 dB
Right channel	5.43 dB

Table 2. Additional average prediction gains obtained with the cascaded NLMS filters used in the second stage

4.2. Compression ratios

We compared the compression ratios of four lossless codecs. Two state-of-the-art codecs, LPAC [2] and Monkey's Audio [3]. A third one, Codec A, based only on linear prediction [6]. And the algorithm proposed in this paper, named Codec B. LPAC uses forward linear prediction and is the predecessor of the MPEG-4 ALS. Monkey's Audio uses multiple passes of several linear predictors with small orders and fixed parameters. Both coders are set to the maximum compression mode. The tests were performed with 6 tracks of different styles of music and also an MPEG test base. Table 3 shows that Codec B outperformed Codec A. Moreover, the performance of Codec B is equal or superior to the best state-of-the-art coders.

Algorithm	LPAC	Monkeys	A	B
MPEG test	2.503	2.573	2.381	2.720
Born in the USA	1.424	1.461	1.418	1.461
Concerto	1.924	1.998	1.872	2.000
Cosmic girl	1.521	1.578	1.540	1.578
Luka	1.422	1.490	1.452	1.491
Polonaise	2.610	2.700	2.284	2.701
Training	1.582	1.621	1.568	1.617

Table 3. Comparison of the compression ratios

5. CONCLUSION

We have proposed in this paper a lossless audio coder whose performance is equal or superior to the best state-of-the art coders. Our coder operates in two stages. The first stage is based on a backward joint-channel prediction that uses, for each frame and for each channel, either an MLP neural network or a linear predictor, depending on which approach gives the best prediction gain. We have shown that this type of prediction clearly outperforms the backward joint-channel linear prediction used in [6]. The second stage is based on two cascaded NLMS filters applied to the residual signal of each stereo channel. As the residuals are not fully decorrelated, this filtering removes the remaining redundancy from the prediction errors. We have shown that this second stage greatly improves the performance of the first stage.

Possible future works would be to investigate other neural network architectures, such as radial basis function networks (RBF) or pipelined recurrent neural networks (PRNN).

6. REFERENCES

[1] T. Liebchen, "The mpeg-4 als homepage (software)," http://www.nue.tu-berlin.de/forschung/projekte/lossless/mp4als.html.

[2] T. Liebchen, "The lpac homepage (software)," http://www.nue.tu-berlin.de/wer/liebchen/lpac.html.

[3] "Monkey's audio homepage (software)," http://www.monkeysaudio.com/.

[4] P. Cambridge and M. Todd, "Audio data compression techniques," in *94th AES convention*, 1993.

[5] T. Liebchen, "Lossless audio coding using adaptative multichannel prediction," in *113th AES convention*, 2002.

[6] Jean-Luc Garcia, Philippe Gournay, and Roch Lefebvre, "Backward linear prediction for lossless coding of stereo audio," in *116th AES convention*, 2004.

[7] M. Faúndez, F. Vallverdú, and E Monte, "Nonlinear prediction with neural nets in ADPCM," in *Proc. ICASSP*, 1998, vol. I, pp. 345–348.

[8] M. Faúndez-Zanuy, "Nonlinear predictive models computation in ADPCM schemes," in *Proc. EUSIPCO*, 2000, vol. II, pp. 813–816.

[9] G. Schuller et al., "Lossless coding of audio signals using cascade prediction," in *Proc. ICASSP*, 2001, pp. 3273–3277.

[10] D.-Y. Huang, "Performance analysis of an RLS-LMS algorithm for lossless audio compression," in *Proc. ICASSP*, 2004, vol. IV, pp. 209–212.

[11] R.F. Rice, "Some practical universal noiseless coding techniques," Tech. rep. jpl-79-22, Jet Propulsion Laboratory, Pasadena, CA, March 1979.