UNIVERSAL SPEECH/AUDIO CODING USING HYBRID ACELP/TCX TECHNIQUES

Bruno Bessette[1,2]*, Roch Lefebvre*[1]*, Redwan Salami*[2]

[1]University of Sherbrooke, Sherbrooke, Quebec, Canada
[2]VoiceAge Corporation, Montreal, Quebec, Canada

ABSTRACT

This paper presents a hybrid audio coding algorithm integrating an LP-based coding technique and a more general transform coding technique. ACELP is used in LP-based coding mode, whereas algebraic TCX is used in transform coding mode. The algorithm extends previously published work on ACELP/TCX coding in several ways. The frame length is increased to 80 ms, adaptive multi-length sub-frames are used with overlapping windowing, an extended multi-rate algebraic VQ is applied to the TCX spectrum to avoid quantizer saturation, and noise shaping is improved. Results show that the proposed hybrid coder has consistently high performance for both speech and music signals.

1. INTRODUCTION

Advances in speech and audio coding have made it possible to significantly reduce the bit rate required to transmit high quality speech and audio. Two major classes of algorithms can be distinguished: linear-predictive (LP) coding, and more specifically CELP coding which is designed to encode primarily speech signals, and transform (or sub-band) coding which is well adapted to represent music signals. These techniques can achieve a good compromise between subjective quality and bit rate for, respectively, speech and music signals. However, neither class of algorithms is universal. Transform coders do not scale well at low bit rates, especially for speech signals. On the other hand, CELP coders, based on an excitation/filter model designed for speech signals, can produce annoying artefacts in the case of music signals. Further, LP-based techniques such as CELP are better adapted to encode low-frequency signals (8 or 16 kHz sampling frequency) and are less adapted to encode fine spectral details in full-band signals ($\geq$ 32 kHz sampling rate).

Emerging applications such as wireless audio streaming, multimedia messaging and multimedia broadcast in cellular networks require both low bit rates and a good quality over all signal types including speech, music, speech between music, and speech over music. This translates into audio compression technology that is both low bit rate and signal independent. Several approaches have then been considered to encode general audio signals (including both speech and music) with a good and fairly constant quality. Transform predictive coding (TPC) [1] is one example that integrates LP and transform coding techniques into a single framework. In TPC, the prediction residual is quantized in the frequency domain, using either open-loop or closed-loop optimization. Hence, compromises have to be made between time and frequency resolution to optimize both the prediction gains and the quantization performance in the transform domain.

A more general approach is multi-mode coding, where each audio frame can be encoded using one of several coding modes. One solution, as in [2], is to use open-loop signal classification to select, for each audio frame, one among N different encoders (for example a CELP coder and a transform coder). In theory, the optimal encoder can be selected automatically for each frame, depending on the signal class. However, designing a robust signal classifier for this purpose is a difficult task, and switching between independent coders can produce audible artifacts.

Another solution, as was proposed in [3], is to integrate CELP coding and transform coding in a single hybrid encoder. In this model, the excitation signal is either encoded using an ACELP codebook, or by applying VQ to the FFT of the target signal. In the ACELP/TCX coder described in [3], 20-ms frames were used with non-overlapping rectangular windows in TCX mode.

In this paper, we extend the work presented in [3] to improve the audio quality in particular for music. Longer, 80-ms super-frames are used with adaptive and overlapping windowing. Further, extended algebraic VQ is used to avoid quantizer saturation in TCX mode. Noise shaping is also improved by subjecting the spectral quantizer to a pre/post processing.

The paper is organized as follows. Section 2 presents an overview of the hybrid ACELP/TCX encoder. The windowing and closed-loop mode selection is presented in section 3. Section 4 describes the spectral quantization in TCX mode. Section 5 gives the results of a subjective (MUSHRA) test. Finally, Section 6 gives some conclusions.

2. ENCODER OVERVIEW

The encoder uses the same basic structure as described in [3], except that longer frames and overlapping windows are used, and more encoding modes are possible by allowing several TCX frame lengths. The input signal $s(n)$ is split into 80-ms super-frames. Each super-frame can be divided in 20, 40 or 80-ms frames. ACELP, and more specifically AMR-WB [4], can be used in any 20-ms frame. Alternatively, TCX with algebraic VQ can be used in each 20-ms frame, or in 40-ms frames or in a single 80-ms frame (i.e. for the whole super-frame). To simplify mode selection, 40-ms frames are only possible by grouping the first two, or the last two 20-ms frames of the super-frame. Hence, there are 26 different mode combinations within an 80-ms super-frame. The first combination corresponds to having a single 80-ms TCX frame in the super-frame. The second combination corresponds to two consecutive 40-ms TCX frames. Then, 8 additional combinations correspond to a single 40-ms TCX frame at either the beginning of the end of the super-frame, with two 20-ms frames (each in either ACELP or TCX) in the rest of the super-frame. The last 16 combinations correspond to all 20-ms frames, each in either ACELP TCX mode. Mode selection can be performed in either open-loop or closed-loop. Closed-loop mode selection will be described in Section 3.

The ACELP and TCX modes are integrated in the sense that they both rely on LP analysis and excitation coding. In ACELP, the excitation is encoded using a sparse codebook in the excitation domain, whereas in TCX the codebook is in the target, or weighted signal, domain. In the proposed ACELP/TCX model, LP analysis is performed every 20 ms, using a half-sine window positioned at the middle of the first 5-ms sub-frame in the next frame. The LP coefficients are quantized and transmitted at different update rates. In ACELP modes, the LP coefficients are transmitted at every 20-ms frame, as in AMR-WB [4]. In TCX mode, the LP coefficients are transmitted once per TCX frame (20, 40 or 80 ms).

3. ADAPTIVE WINDOWING AND MODE SELECTION

3.1. Windowing in TCX mode

To allow smooth mode transitions, and to reduce blocking effects, overlapping windows are applied to the target signal. One difficulty is that ACELP must use a rectangular window, whereas a transform coding mode such as TCX has better performance using overlapping windows. The window shape must then be chosen to simultaneously improve transform coding performance and allow smooth transition from an ACELP frame to a TCX frame. For this purpose, we propose the following window shape to apply to the weighted signal when encoding in TCX mode. The window is formed by the concatenation of three sub-windows:

$$w_1(n) = \sin(2\pi\, n / (4\, L_1)) \quad \text{for } n = 0, \ldots, L_1\text{-}1$$
$$w_2(n) = 1 \quad \text{for } n = 0, \ldots, L\text{-}L_1\text{-}1$$
$$w_3(n) = \sin(2\pi\, n / (4\, L_2)) \quad \text{for } n = L_2, \ldots, 2L_2\text{-}1$$

The constants L_1, L and L_2 control, respectively, the shape of the left, middle and right part of the window. The window is a compromise between a rectangular window and an overlapping half-cosine window. Given the internal sampling rate of 12.8 kHZ in AMR-WB, each 10-ms duration corresponds to 128 samples (1024 samples for the whole 80-ms super-frame). Then the values of L, L_1 and L_2 are chosen as follows. L_1, which controls the overlap duration in the left-part of the window, is set to 0, 32, 64 or 128 when the previous frame was, respectively, 20-ms ACELP, 20-ms TCX, 40-ms TCX or 80-ms-TCX. The values of L and L_2 are set as a function of the TCX frame length. In 20-ms TCX, L = 256 and L_2 = 32. In 40-ms TCX, L = 512 and L_2 = 64. Finally, in 80-ms TCX, L = 1024 and L_2 = 128. Note that L_2 controls the overlap duration in the right portion of the window, and thus corresponds to a look-ahead into the next frame. Figure 1 illustrates the two possible window shapes in TCX mode depending if the previous frame was ACELP (upper curve) or TCX (lower curve).

The cosine parts in the window in Figure 1 correspond to the overlap. When two consecutive TCX frames occur, frame transition is smoothed by this overlap. When a TCX frame follows an ACELP frame, the transition is handled differently. The zero-input response (ZIR) of the weighting filter ($W(z)$ in AMR-WB) is first computed and truncated. Then, the truncated ZIR is subtracted from the first samples of the weighted signal. Finally, multiplying by the proper window gives a signal which tends to zero in both ends, as if multiplied by the symmetrical window of curve (b) in Figure 1.

Since TCX quantizes the target in the FFT domain, this smoothing effect will reduce the noise floor in the transform domain and thus improve quantization. At the decoder, the truncated ZIR will simply be added to the inverse transform of the decoded spectrum, to recover the quantized weighted signal. Note that the window shape in TCX is a compromise.

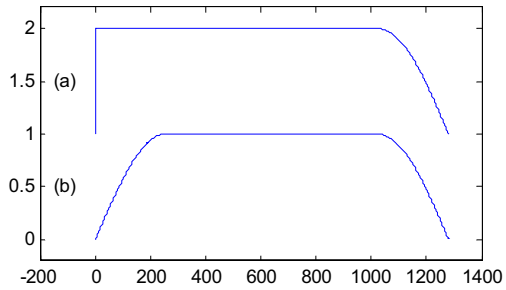

Figure 1. TCX window shape example when (a) previous frame is ACELP and (b) previous frame is TCX.

The spectral characteristics of this window are not as optimal as, for instance, a Hanning window. However, the spectral dynamics of the *weighted* signal are significantly lower than the spectral dynamics of the original signal so using a window with a higher noise floor is less critical.

3.2. Closed-loop mode selection

As described in Section 2, there are 26 different mode combinations in an 80-ms super-frame. However, each 20-ms frame can only be in one of four modes, as shown in each column of Figure 2, where Fr1 to Fr4 are the first, second, third and fourth frame in the super-frame. In the figure, "A" represents ACELP mode, and "T" represents TCX mode (with T20 standing for 20-ms TCX, and so on). Then, a closed-loop mode selection taking advantage of this redundancy and involving only 11 "Trials" is summarized by Figure 2.

Trial	Fr1	Fr2	Fr3	Fr4
1	A			
2	T20			
3		A		
4		T20		
5	T40	T40		
6			A	
7			T20	
8				A
9				T20
10			T40	T40
11	T80	T80	T80	T80

Figure 2. Encoding mode tried in each 20-ms frame (Fr1 to Fr4) for each of the 11 trials of the closed-loop mode selection.

In Trials 1 and 2, the first 20-ms frame is encoded in ACELP then in TCX, and the best mode for this first frame is selected temporarily. Mode selection is performed according to the average segmental SNR in the weighted signal domain (target signal in a speech coder), using 5-ms segments for SNR computation. Then, applying proper windowing as in Section 3.1 and using the temporary mode decision for Fr1 (after Trial 2), the second 20-ms frame is encoded in ACELP then in TCX in Trials 3 and 4. A temporary mode decision is also taken for Fr2, using average segmental SNR. Then, in Trial 5, Fr1 and Fr2 are

grouped to form a 40-ms frame which is encoded as a whole in TCX mode. After Trial 5, the temporary mode decision for Fr1 and Fr2 is either 40-ms TCX or the modes retained after Trials 2 and 4. Again, average segmental SNR is used in the selection, computed on the complete duration of the mode (here, either 20 ms or 40 ms). Trials 6 to 10 repeat this temporary mode selection for Fr3 and Fr4. Finally, the results are compared to encoding the whole 80-ms super-frame in a single 80-ms TCX frame (Trial 11). After Trial 11, a final mode decision can be made for all four 20-ms frames.

4. TCX SPECTRUM QUANTIZATION

As described in Section 2, the ACELP frames are encoded using the AMR-WB encoder [2], with small modifications on the LP analysis window to better integrate with TCX coding. In TCX modes, the signal is encoded as in [5] applying transform coding to the weighted signal. Frame windowing and overlapping is as described in Section 3.1. A fast Fourier transform is used to map the signal to the frequency domain, and lattice quantizers are used to encode the spectral coefficients.

The spectral coefficients are quantized by grouping four consecutive complex-valued coefficients to form 8-dimensional real-valued vectors. As in [5], a lattice quantizer based on the RE_8 lattice [6] is used. Codebooks of different bit rates are constructed by selecting subsets of appropriate size from the RE_8 lattice points. The codebooks, labeled Q_n, comprise 2^{4n} codevectors. It was found more efficient not to use the ½ bit per sample codebook Q_1, so the allowed codebooks are Q_0 (the null vector), Q_2, Q_3, Q_4, etc. at 0, 1, 1.5, 2, etc. bits per sample respectively. Any lattice point can be described as the component-wise permutation of a so-called *leader*; using very few absolute leaders (i.e. with all positive components), one can generate thousands of lattice points by position and sign permutations of the components. Hence, storage for a given codebook Q_n is essentially limited to a few 8-dimensional vectors – the leaders. The specification of Q_0, Q_2, Q_3 Q_4 is as described in [5].

To avoid quantizer saturation, an extension is applied to generate multi-rate codebooks of sizes much larger than Q_4. Specifically, a method called the *Voronoi extension* allows the construction of larger codebooks by scaling a base codebook (here, Q_3 or Q_4) by an integer multiple of 2, and filling the Voronoi regions around the scaled codebook with a sub codebook called the extension codebook. A lattice point can then be described as the sum of a codevector in the scaled codebook, and a codevector in the extension codebook. It is obvious that applying the Voronoi extension generates codebooks which extend higher up in 8-dimensional space, and that these codebooks comprise more and more lattice codevectors as the extension scaling factor increases. The details of the Voronoi extension can be found in [5]. The extension increases the codebook size by ½ bit per sample at a time.

For each 8-dimensional vector, the information sent to the decoder is 1) the codebook index n, 2) the lattice point index selected in Q_n – described by a leader and a permutation and 3) a Voronoi index if the extension was applied – i.e. if a codebook larger than Q_4 was used. A few bits are also used to send the global gain of the TCX frame (typically around 6 or 7 bits).

Using these lattice codebooks with the Voronoi extension, the TCX frame (20, 40 or 80 ms) is quantized for minimum mean-squared error (MSE). Since the source vectors have to "fit" into the lattice codebooks, an iterative gain-shape approach has to be used, in principle, by applying the steps of Figure 3.

This would require quantizing the frame several times before obtaining the optimal gain. Hence, in practice the gain is estimated without going into the quantization loop of Step 2. It can be shown experimentally that, when quantizing a source vector x_k with energy E_k, the number of bits required to transmit the selected codevector in Q_n and the Voronoi extension index is well approximated by

$$R_k = 4\log_2\left(\frac{E_k}{2}\right) \quad (1)$$

1) Divide the TCX frame by a scaling factor g
2) Quantize each 8-dimensional sub-block of transform coefficients using the lattice codebooks Q_0, Q_2, Q_3 Q_4 and possibly the extension
3) Verify if the bit consumption for each 8-dimensional block meets the bit budget (as close as possible without exceeding it)
4) Select another scaling factor g and go back to Step 1 if the condition in Step 3 is not met

Figure 3. Principle of quantization with lattice codebooks.

A simple iterative loop can be devised to find a global gain g which ensures that the sum of the R_k for each scaled 8-dimensional vector in the spectrum meets the bit budget. After estimating the gain, only the first two steps of Figure 3 need to be applied, removing the need for a costly closed-loop iterative approach.

The lattice VQ applied to the TCX spectrum distributes the quantization noise uniformly in the whole spectrum. Since the signal quantized is the weighted signal, and not the original signal, this coding noise will be shaped by the inverse of the weighting filter. This coarse noise shaping may be adequate for speech, but general audio requires a finer shaping especially at low frequencies. For this, we use an adaptive pre/post processing approach that is applied before and after quantization of the spectral coefficients. The post-processing is applied at the decoder, but also at the encoder since the excitation signal is necessary to encode the next frames because of the predictive nature of the encoder. Experiments have shown that applying the pre/post processing up to 1 or 2 kHz is subjectively optimal. More precisely, we restrict the pre/post processing to the first quarter of the spectrum which translates into 0-1600 Hz at 12.8 kHz sampling.

First, the pre-processing operates as follows. The complex-valued spectral coefficients of the TCX spectrum X are grouped in blocks of 4 consecutive coefficients corresponding to the 8-dimensional real-valued vectors quantized by the lattice VQ. Since spectrum X has $K = L/2$ complex values, where L is the TCX window length, there are $M = K/4$ 8-dimensional blocks B_m with $m = 0, \ldots, M\text{-}1$. The spectral coefficient at Nyquist frequency is set to zero. Then, for each 8-dimensional block B_m in the first quarter of the spectrum, the block energy E_m is calculated (sum of the squares). With E_{max} being the maximum block energy, the spectral pre-processing operates as in Figure 4 for $m = 0, \ldots, (M/4)\text{-}1$.

Step 1 : For block B_m, calculate the ratio
$$R_m = (E_{max} / E_m)^{0.50}$$

Step 2 : if $R_m > 10$, then set $R_m = 10$
Step 3 : also, if $R_m > R_{m-1}$ then set $R_m = R_{m-1}$
Step 4 : Modify block B_m by the final ratio R_m i.e.
$$B_m = R_m * B_m$$

Figure 4. Pre-processing in TCX spectrum.

Step 4 in Figure 4 ensures that the ratio function R_m decreases monotonically. Further, limiting the ratio R_m to be smaller or equal to 10 means that no spectral components in the low-frequency emphasis function will be modified by more than 20 dB.

In the post-processing which is applied after quantization of the spectral coefficients, the initial ratio calculation is Step 1 of Figure 4 is replaced by

$$R_m = (E_{max} / E_m)^{0.25}$$

and the multiplication of B_m by R_m of Step 4 is replaced by the division of B_m by R_m. It can easily be shown that the post-processing is exactly the inverse process as the pre-processing, so that taking the unquantized spectrum would yield exactly the FFT input at the FFT output.

Applying this pre-post processing technique to the TCX spectrum prior to and after quantization has the effect of shaping the coding noise in the lower band (0-1600 Hz), such that in particular low energy components before the first spectral peak will be better encoded.

5. PERFORMANCE EVALUATION

The subjective performance of the proposed hybrid ACELP/TCX coder at 24 kbps was evaluated in a MUSHRA test [7]. The following reference codecs were included in the test: G.722.1 at 24 kbps (transform based optimized for music), G.722.2 (AMR-WB) at 23.85 kbps (ACELP based optimized for speech), and G.722 at 48 and 64 kbps (generic codec at high rates). It should be noted that the presented hybrid codec can operate at a bandwidth higher than 7 kHz through the use of bandwidth extension and/or resampling the signal at the encoder input to rates higher than 12.8 kHz. This will results in significantly improved quality over 7 kHz wide signals. However, in the conducted test the bandwidth was normalized to 7 kHz for all samples, to force the listeners to concentrate on coding artifacts rather than fullness of sound.

Eight experienced listeners participated in the subjective evaluation, not involving the codec developers. The audio material included male and female speech, instrumental music, and music with vocals. The results are summarized in Figure 5.

Several observations can be made. First, as discussed in the introduction, the transform coder (G.722.1) and the LP-based coder (G.722.2) have inconsistent quality, depending on the input signal type. However, the quality of the hybrid ACELP/TCX codec is consistently good over all signal types included in the test. Compared to G.722.2 (AMR-WB) the music quality is significantly improved. Further, the quality of the proposed hybrid coder at 24 kbps is equivalent to or better than G.722 at 64 kbps for both music and speech. In the case of speech, the improvement of the hybrid coder over G.722.2 at the same rate can be explained by the more flexible frame structure. Further, the low-frequency noise shaping in TCX mode helps reduce inter-harmonic noise in particular for female speech.

6. CONCLUSION

We have presented a hybrid audio coder based on the integration of an LP-based model (specifically, the AMR-WB speech coder) and a transform coding model (TCX).

The coder uses fixed frame length (20 ms) in LP-based mode, and variable frame length (20, 40 or 80 ms) in transform coding mode. Overlapping windows are used in TCX modes. The weighting filter zero-input response is used to handle ACELP-TCX transitions. Algebraic structures are used to quantize the excitation signal – ACELP codebooks in LP-based mode, and lattice VQs in transform coding mode. By using longer delays and integrating several coding modes, the proposed hybrid codec has consistently good quality for different content, including speech, music and speech over music. This hybrid coding paradigm has been used in the AMR-WB+ codec recently selected by 3GPP for multimedia messaging and packet-switched streaming services.

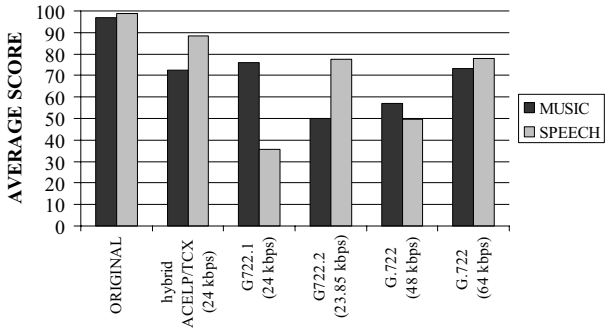

Figure 5. Summary of MUSHRA test results.

7. REFERENCES

[1] J.-H. Chen and D. Wang, "*Transform predictive coding of wideband speech signals*," Proceedings ICASSP-96, vol. 1, 7-10 May 1996, pp. 275-278

[2] L. Tancerel, S. Ragot, V.T. Ruoppila, and R. Lefebvre, "Combined Speech and Audio Coding by Discrimination", *IEEE Workshop on Speech Coding*, Delevan, Wisconsin, U.S.A., pp. 158-160, September 17-20, 2000

[3] B. Bessette, R. Salami, C. Laflamme, R. Lefebvre, "*A wideband speech and audio codec at 16/24/32 kbit/s using hybrid ACELP/TCX techniques,*" Proceedings IEEE Workshop on Speech Coding, Porvoo, Finland, pp. 7-9, June 20-23 1999.

[4] 3GPP TS 26.190, "3rd generation partnership project, technical specification group services and systems aspects, speech codec speech processing functions, AMR wideband speech codec; transcoding functions"

[5] S. Ragot, B. Bessette and R. Lefebvre, "Low-complexity multi-rate lattice vector quantization with application to wideband speech coding at 32 kbit/s", Proc. IEEE ICASSP, Montreal, Canada, pp. I-501 to I-504, May 2004.

[6] J.H. Conway and N.J.A. Sloane, "A fast encoding method for lattice codes and quantizers", *IEEE Trans. Inform. Theory*, vol. 29, no. 6, Nov. 1982, pp. 820-924.

[7] Recommendation ITU-R BS.1534, "Method for the subjective assessment of intermediate quality level of coding systems".