Tree Encoding for the ITU-T G.711.1 Speech Coder

Abdul Hannan Khan and *Peter Kabal*

Electrical & Computer Engineering, McGill University, Montreal, Canada
abdul.h.khan@mail.mcgill.ca, peter.kabal@mcgill.ca

Abstract

This paper examines enhancement to ITU-T Recommendation G.711.1 PCM wideband extension speech coder. To further improve the core lower-band coding performance the use of vector quantization and delayed decision coding is studied. A particular case of delayed decision coding, tree encoding, is implemented in the above standard. The bitstream is compatible with both the legacy G.711 and the G.711.1 decoder. PESQ (ITU-T P.862, Perceptual Evaluation of Speech Quality) is used to evaluate the performance. Both the vector quantizer and tree encoder have better performance than the original core layer encoder.

Index Terms: speech coding, G.711.1, tree encoding

1. Introduction

IP telephony is becoming increasingly common in the telecommunication industry. Telecommunication service providers are moving towards an all IP network. One such coder is the recent ITU-T G.711.1 extension to the ITU-T G.711 PCM coder. The G.711.1 extension adds noise feedback and a lower-band enhancement layer, as well as a wideband encoding layer. The noise feedback applies perceptual masking to the quantization noise introduced by the PCM quantizer. The perceptual filter is based on a linear prediction (LP) analysis. The enhancement layer allows more bits to be used for encoding, hence, increasing the number of quantization levels. This reduces the quantization noise at the expense of a higher bit rate. A higher band encoding option is also available for wideband telephony. This paper deals only with the lower-band coding, which is also used as part of the wideband option. This paper examines the incorporation of vector quantization (VQ) and delayed decision multi-path tree encoding. The delayed decision multi-path tree encoding is implemented by the (M,L)-algorithm, where M is the maximum number of tree paths available after quantizing a block of input samples and L is the maximum depth of the tree. The parameter L also sets the coding delay. Because the noise feedback filter has memory, current outputs affect future decisions. The new quantization strategy takes into account past history. The final bitstream is compatible with both the legacy G.711 and the G.711.1 decoder.

2. Background

In 2008, ITU-T standardized G.711.1 [1] which is an extension to G.711. The new coder has an embedded structure and is backward compatible with legacy G.711 decoder. The legacy G.711 log companded PCM encoder codes the telephony band 300–3400 Hz at 64 kb/s. The new standard has several options. Here we consider the narrowband coding options at 64 kb/s and 80 kb/s. The coder extends the bandwidth over that of legacy G.711 and reduces the effect of quantization noise with noise feedback and (at 80 kb/s) increased sample resolution.

- Layer 0: 64 kb/s, included noise feedback at the encoder
- Layer 1: Lower-band enhancement layer; optional (16 kb/s additional)
- Layer 2: Upper-band enhancement layer; optional (16 kb/s additional)

The core layer, at 64 kb/s, is compatible with the legacy G.711 decoder.

2.1. G.711.1 Core Layer

In this paper we will be consider a μ-law quantizer. It encodes a 16-bit sample into an 8-bit code (sign, exponent (3 bits) and mantissa (4 bits)). The decoder maps the 8-bit codes back to 16-bit values by table lookup. This is what is used in the legacy G.711 coder. The G.711.1 coder uses noise feedback and dead-zone quantizer.

2.2. Noise Feedback

The G.711.1 encoder has a local decoder which is used to determine the quantization error. The difference between the decoder output and the input signal is filtered and added to the next input sample. This perceptual filtering makes use of the properties of the human perception system and tends to masks the quantization noise. The perceptual noise shaping filter is based on a LP filter, and is given by [2],

$$F(z) = A(z/\gamma) - 1, \tag{1}$$

where $A(z)$ is a fourth order LP prediction error filter and γ is a perceptual weighting factor.

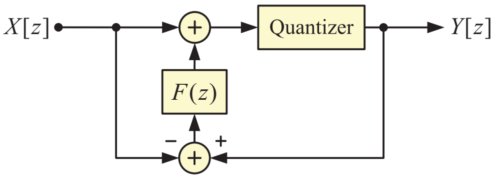

Figure 1: Noise shaping quantizer

From Fig. 1 (see also [3]),

$$Y(z) = X(z) + \frac{Q(z)}{1+F(z)}, \tag{2}$$

where $Q(z)$ is the quantization noise added at the quantizer, $X(z)$ is the input signal, and $Y(z)$ is the locally decoded signal. It can be seen that the spectrum of quantization noise is shaped with the spectrum of $1/A(z/\gamma)$. The noise shaping filter is adaptive to the incoming signal and updated frame-by-frame (every 40 samples).

2.3. Dead-Zone Quantizer

G.711.1 adds another feature to the core layer – the quantizer implements a dead-zone. The dead-zone quantizer affects low energy signals. The normalized lowest quantization outputs in a μ-law quantizer are 0 and ± 8. Very low level signals, like those of ambient noise, can get quantized to the ± 8 level. For low energy frames, the dead-zone quantizer becomes active and sets the output to zero. Details of the operation can be found in [1].

3. Tree Encoding

A vector quantizer takes a block of input samples and quantization them together. When noise feedback is included, the effect of previous decisions propagates beyond the block. Delayed decision coding can be used to take the noise propagation into effect. With delayed decision coding, one considers the effect of possible current decisions on future samples. In [3] it is shown that ADPCM with vector quantization is similar to a CELP coder. The G.711.1 core layer is similar to ADPCM in the sense that both are waveform coders. Reference [4] is an examples of tree encoding in a CELP coder. Building on these tree encoding will be introduced in G.711.1 core layer.

3.1. Single Path Tree Encoding

Three important terms are associated with tree encoding – nodes, branches and leaves. A node is a block of samples which has a quantizer output associated to it (see Fig. 2). If the block includes more than one sample, the quantizer implements vector quantization (VQ). For a single path tree encoder a tree is only left with one node once a decision has been made. Whenever a new block is received and a decision has been made, the tree branches from this node. At the end of each branch is a leaf. Each leaf corresponds to a possible quantizer output value. Once the best possible match has been selected, the end of the selected leaf becomes the node for the next round and the rest of the leaves are discarded. As new blocks are processed, the tree is continuously populated and pruned. This type of coding can be seen in CELP; the coder takes into account the propagation effects of noise feedback due previous blocks, but does not consider the effect on samples beyond the block. If the vector quantizer is replaced by a scalar version, the single path tree is also a model for simple PCM encoders.

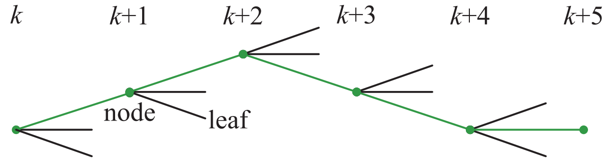

Figure 2: Single path tree encoding for a branching factor of 3

3.2. Multi-path Tree Encoding

If the decision is held off until its effect on further decisions can be evaluated, multi-path tree encoding is realized. The tree is branched from multiple nodes and, therefore, many more leaves are available to choose from. The (M,L)-algorithm is used to implement the multi-path tree encoder [5][6]. This algorithm is defined by the two parameters M and L: M is the maximum number of nodes to be retained after a decision has been made; L is the depth of the tree.

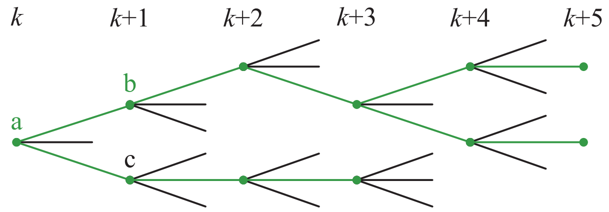

Figure 3: Multi-path tree encoding for a branching factor of 3, $M = 2$, $L = 3$

After each block has been processed, a maximum of M nodes are kept. A maximum specified delay is allowed and if the retained paths have not converged by then, a decision is forced. The code for that block is transmitted. At the next instance, when the next input block arrives, each of the M nodes is populated with a number of leafs equal to to the size of the codebook. Compared to the single path tree encoder, M times more output choices are available. Each path has its own error associated with it, and the filter states on each path are different as well. Once the nodes have been populated, the leaf with the best quantization output associated with it according to the cumulative error criterion, to be described later, is chosen. Once this selection is done, the path is traced $L - 1$ nodes back and the node which leads to this selected leaf is chosen as the best code for the input block $L - 1$ nodes in the past. Therefore, a delay of $L - 1$ is created. After this, the tree is pruned and the best M paths are selected for retention. Only paths that emanate from the oldest node are kept when pruning the tree. This encoding process continues as further blocks are input.

In the particular example of Fig. 3, $M = 2$, $L = 3$ and codebook size of 3 is chosen. At time $k+1$ the tree is branched from node a with a branching factor same as the size of the codebook, which is 3. The tree is pruned and, based on the error criterion, the best M number of nodes are kept behind. At time $k + 2$ each of these nodes is further branched out and the resulting tree is again pruned. This is repeated till time $k + 3$ when a decision has to be made between node b and c for the input block corresponding to time $k+1$. Before the tree is pruned the best node at time $k+3$ is selected and the path is traced back to time $k + 1$. In this particular example the path is traced back to node b. To maintain continuity only paths emanating from node b are kept, as in Fig. 3. The process continues till all the input blocks have been encoded.

The maximum number of nodes in a tree, for a depth of L is $2^{m(L-1)}$. Therefore,

$$M \leq 2^{m(L-1)} \tag{3}$$

There are two special cases of multi-path tree encoding. The first one is when $L = 1$. In this case $M = 1$ as well and single path tree encoding is realized. The other special case is when $M = 1$. In this case only one node is retained after the decision has been made. There is only one path and increasing the tree depth would only add delay without any benefits.

3.3. Cumulative Error

The error measure decides how the tree is populated and in turn pruned. Hence, it plays a vital role in tree encoding. The cumulative error over the whole path is chosen to be the error measure to make use of a tree encoder's ability to consider future values. The cumulative sum of the mean square error of all nodes in the path is considered. At time instant $k + 1$, a decision is made for the code for the input block at time instant $k-(L-2)$. As all the paths originate from the already chosen node at time $k-(L-1)$ the cumulative error till that point is common to all paths. Hence, the part of the cumulative error up to time $k-(L-1)$ is discarded. This helps in keeping the cumulative error from continuously growing.

$$E_{\text{opt}} = \min_{j}\Big[\sum_{l=k-(L-2)}^{k+1} e_j^2(l)\Big], \qquad 0 \le j \le M-1. \quad (4)$$

3.4. Modification To G.711.1 Core Layer

By replacing the quantizer with a codebook based VQ, the G.711.1 core layer looks like Fig. 4. The best codebook entry is chosen by considering the error from MSE block. Conceptually, each codebook entry is evaluated and associated with a weighted mean-square error. The entries with the lowest errors are retained.

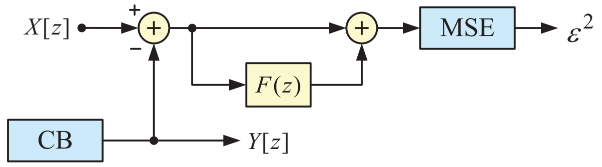

Figure 4: G.711.1 core layer using a VQ codebook

The modifications to the G.711.1 core layer look like the tree in Fig. 3 with each new leaf having a VQ encoder like that shown in Fig. 4. Each node under consideration keeps track of the filter memory, the smallest cumulative error up to that node and VQ entry associated with it. Each leaf emanating from a node, calculates the incremental increase in error and keeps track of the VQ entry. The (M,L)-algorithm is used to prune the tree and force decisions after a delay of L. The dead-zone quantizer used in G.711.1 is not used for tree coding. In fact, it is counter-productive since values which are forced to zero will have a larger error and may be ignored by the delayed decision process.

4. Experimental Results

4.1. Complexity

A PCM quantizer has 256 possible output levels for each sample. We restrict the block size to 2 samples to restrain the number of VQ codebook entries. The codebook search is performed in the local neighbourhood of the input samples. This neighbourhood is chosen to be ±2 levels from the default quantized values. With a block size of 2, the codebook size is 25. In the G.711.1 core layer there are two major operations, quantization and filtering. With a vector quantizer there is a single quantization operation but the filtering is applied to each element of the 25 entries of the codebook subset. In a tree encoder the number of filtering operations is now M-times the size of the codebook sub-set. The filtering operation requires 4 multiplications and 3 additions per sample. After each filtering operation, the mean square error is calculated. Each mean square error calculation for two samples requires 2 multiplications and 3 additions.

For $M = 3$, the increase in complexity for tree encoding relative to scalar quantization is substantial. However G.711.1 needs processing power for the lower-band enhancement layer and the higher-band layer. When working at 64 kb/s only the core layer is present. Tree encoding for the core layer can make use of the otherwise idle computational resources.

4.2. Performance

Four speech sentences were used to test the coding schemes. These were recited by two different speakers, one male and one female. An objective measure, Mean Opinion Score (MOS) as measured by PESQ (Perceptual Evaluation of Speech Quality, ITU-T P.862 [7]), was used to evaluate the speech quality of the various systems. One expects that the quantizer modified with codebook based VQ and delayed decision coding implemented by the (M,L)-algorithm should perform better than the original G.711.1 core layer quantizer. To evaluate this comparison has been made at three different signal levels. In the first scenario the signal is used in without any attenuation – the signal occupies most of the dynamic range of the quantizer. In the second and third scenarios, the power of the signal is attenuated by 20 and 40 dB to mimic quite talkers and to force the quantizer to use fewer quantization levels. The PESQ scores for the three cases under the three scenarios are listed in Table 1. The entry for the tree encoder uses $M = 3$ and $L = 3$. As we will see in a later section, the performance has saturated for these values. Scores for the G.711.1 core layer with the added lower-band enhancement layer and legacy G.711 are provided as well.

Table 1: PESQ MOS scores for G.711.1 encoders at various attenuation levels

Encoder	Rate	0 dB	20 dB	40 dB
G.711	64 kb/s	4.27	3.61	2.56
Core Layer	64 kb/s	4.25	3.65	2.26
Core + VQ	64 kb/s	4.31	3.69	2.62
Core + Tree	64 kb/s	4.31	3.68	2.63
Core + Enh.	80 kb/s	4.42	4.28	3.31

At 64 kb/s, the tree encoder and vector quantizer provide the best objective results, better than the G.711.1 core layer. For the attenuated signals, listeners were able to clearly rank the results in the same order as the PESQ scores. The low score of G.711.1 core layer at 40 dB attenuation can be partly attributed to the dead-zone quantizer which zeros the output for low level signals. The lower-band enhancement layer increases the number of quantization levels available. This provides an increase in performance (Table 1), but at the cost of an increased data rate. This performance increase is more noticeable in the case of the attenuated signal. The enhancement layer garners the benefit of the increased number of quantization levels. Tree encoding and vector quantization take a step towards reaching this performance level and retain the base data rate of 64 kb/s.

In informal subjective testing, for the signal without attenuation, at 64 kb/s, the encoders gave essentially indistinguishable results. For the attenuated signal, the modified encoders provide better quality. The speech is less broken. As was the case with the PESQ scores, with the addition of the lower-band enhancement layer, at a total rate of 80 kb/s, the increase in subjective quality is the largest.

4.2.1. Tree coder performance as a function of M

An increase in M means more nodes are kept back after each decision instance. Hence, more leaves are available when a new input signal is received. The performance of the system as a function of M for a signal attenuation of 40 dB is plotted in

Fig. 5. The performance of G.711.1 core layer is provided for reference. The value of L is kept constant at 6. The performance increases as the vector quantizer is turned on, but it saturates quickly for the tree encoder. The point for $M = 1$ is effectively just 2-sample VQ. Subjectively there is not much difference between the different tree encoded signals.

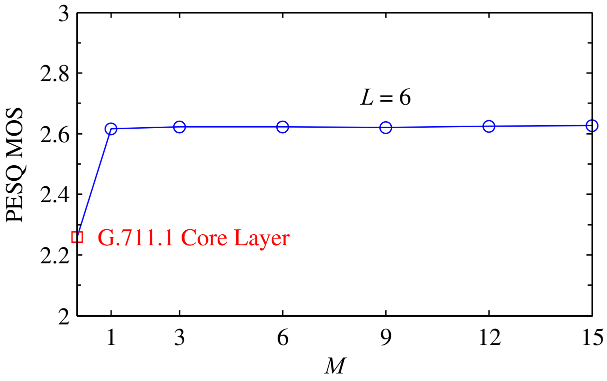

Figure 5: PESQ MOS as a function of M for an attenuation of 40 dB. The first point is for the G.711.1 core layer.

In the case of a μ-law quantizer, where the quantizer has a large range and large outer quantization intervals, there will be a lot of codewords which do not give satisfactory error results when used to approximate the input block. With a smaller M these tend to get eliminated very quickly. Only the good approximations are kept. The benefit can be seen by the sudden increase in the performance of the encoder as compared to the original G.711.1 core layer encoder. The saturation of the performance occurs because when M increases more codewords with bad performance are kept. The tree encoder just ignores these.

4.2.2. Tree coder performance as a function of L

Once $L > 1$, delayed decision multi-path tree encoding is realized. Figure 6 shows the perceptual performance of the tree encoder as a function of L when $M = 6$, at a signal attenuation of 40 dB. As was the case for changes in M, it is seen that there is a sudden increase in performance as the vector quantizer kicks in and then the performance saturates for the tree encoder. The point for $L = 1$ is effectively the same as 2-sample VQ. Note that the noise feedback filter has a filter memory of 4 and the block size is 2. Even with a small value for L, the effect on future samples due to filter memory is taken care of. Once L gets large, the benefit levels off because the major effect of filter memory is only short term. Also, the coarse quantization intervals of μ-law quantizer do not provide enough viable alternatives in terms of selection of different codewords. A further increase in the tree depth only results in an increase of encoder complexity without any performance increase.

5. Discussion and Conclusion

The simulation results show that vector quantization and tree encoding give better results than the G.711.1 core layer. The performance increase is limited by the coarseness of the log quantizer. The VQ implementation used here codes two samples at a time. This configuration reaps most of the benefits to be had. Adding a delayed decision and using multiple parallel candidates in a tree coding structure adds little to the subjective quality. In a G.711.1 coder, computational power is needed when the enhancement layers are used. This computational power can be put to use to implement VQ or tree coding when the enhancement layers are not enabled. The VQ and tree encoding options have two advantages over the use of the lower-band enhancement layer. An increase in quality is achieved without any increase in data rate and the bitstream is fully compatible with legacy G.711 decoders.

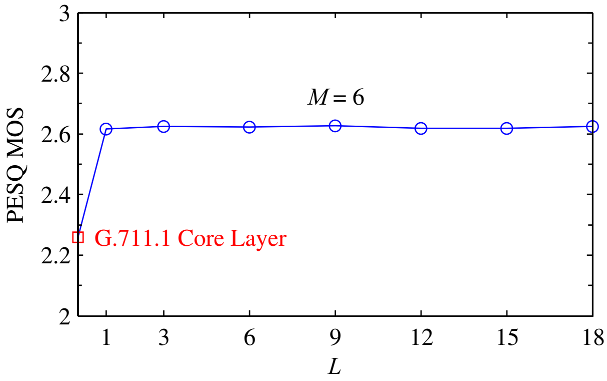

Figure 6: PESQ MOS as a function of L for an attenuation of 40 dB. The first point is for the G.711.1 core layer.

6. References

[1] ITU-T Recommendation G.711.1, *Wideband embedded extension for G.711 pulse code modulation*, March 2008.

[2] J. Lapierre, R. Lefebvre, B. Bessette, V. Malenovsky and R. Salami, "Noise Shaping in an ITU-T G.711-Interoperable Embedded Codec", *Proc. Eur. Signal Processing Conf.* (Lausanne, Switzerland), 5 pp., Aug. 2008.

[3] P. Kabal, *The Equivalence of ADPCM and CELP Coding*, Technical Report, Electrical & Computer Engineering, McGill University, April 2010 (available on-line at `www-mmsp.ece.mcgill.ca`).

[4] V. Iyengar and P. Kabal, "A Low Delay 16 kbit/sec Speech Coder", *IEEE Trans. Signal Processing*, vol. 39, no. 5, pp. 1049–1057, May 1991.

[5] J. B.Anderson and J. B. Bodie, "Tree Encoding of Speech", *IEEE Trans. Inf. Theory*, vol. 21, no. 4, pp. 379–387, July 1975.

[6] C. C. Chu and P. Kabal, "Tree Coding in a Code Excited Linear Predictive Speech Encoder" *Biennial Symp. Commun.* (Kingston, ON), pp. C.3-8–C.3.11, June 1986 (available on-line at `www-mmsp.ece.mcgill.ca`).

[7] ITU-T Recommendation P.862, *Perceptual Evaluation of Speech Quality (PESQ)*, Nov. 2005.