PERFORMANCE AND ERROR SENSITIVITY COMPARISON OF LOW AND HIGH DELAY CELP CODECS BETWEEN 8 AND 4 KBITS/S

J.P. Woodard and L. Hanzo

Dept. of Electr. and Comp. Sc., Univ. of Southampton, SO17 1BJ, UK.
Tel: +44 1703 593125, Fax: +44 1703 594508
Email: jpw@ecs.soton.ac.uk and lh@ecs.soton.ac.uk
http://www-mobile.ecs.soton.ac.uk

ABSTRACT

In this paper we study the performance and error sensitivities of several CELP based codecs operating between 8 and 4 kbits/s. Both forward and backward adaption techniques are used for the short and the long term predictors, and both trained and Algebraic excitation codebooks are used. Three codecs which employ backward adaption of their short term predictors and operate with frame-lengths of 3 ms or less are described. These codecs operate between 8 and 4 kbits/s and their performance at these bit rates is compared to a traditional forward adaptive Algebraic CELP codec operating at 4.7, 6.5 and 7.1 kbits/s. Furthermore, the error sensitivity of the backward adaptive codecs, and means of improving this error sensitivity, are investigated. Finally, we compare the error sensitivity of the low delay, backward adaptive codecs to the high delay, forward adaptive codecs. Surprisingly, we found that it is possible to achieve good error resilience, comparable to that of the forward adaptive codec, using low delay backward adaptive codecs.

1. INTRODUCTION

During the past ten years many speech codecs offering communications to toll quality reconstructed speech at bit rates of 16 kbits/s and below have been developed, and several of these are now commonly used. Until recently most of these codecs used forward adaption to determine the short term linear prediction coefficients, which are used in the encoding and decoding processes. Such codecs typically buffered about 20 or 30 ms of the input speech and used this buffered speech to determine the linear prediction coefficients. However, speech codecs tend to have an end to end delay of about three times the frame-length of the codec [1], and so some of these low rate speech codecs had one-way delays approaching 100 ms.

Such high delays can be undesirable for several reasons. In the public switched telephone network 4 to 2 wire conversions lead to echoes, which will be subjectively annoying, if the echo is sufficiently delayed. Even if echo cancellers are used, a high delay speech codec makes the echo cancellation more difficult. Therefore, if a codec is to be connected to the telephone network, it is desirable that its delay should be as low as possible.

Therefore in recent years much work has been devoted to produce low rate speech codecs with lower delays. In 1992 a low delay 16 kbits/s CELP codec was standardised by the CCITT as G.728 [1, 2]. This codec uses backward adaption to determine the linear prediction coefficients which are used in the encoding and decoding of the speech. This means that the coefficients are derived from the past reconstructed speech, rather than the future input speech, and therefore it is not necessary to buffer a long frame of the input speech for the encoding. This backward adaption allows the G.728 codec to produce toll quality reconstructed speech at 16 kbits/s with a frame-length of only 0.625 ms. More recently an 8 kbits/s codec with a frame-length of 10 ms has been developed [3, 4] and will soon be standardised as G.729. This codec uses forward adaption of the linear prediction coefficients, but manages to maintain a reasonably low delay by using a frame-length of only 10 ms, along with vector quantization of the coefficients.

In this paper we seek to compare the performance and error robustness of four CELP based speech codecs operating between 4 and 8 kbits/s. Initially in Section 2 we describe three backward adaptive codecs, based on the philosophy of the G.728 codec, which operates at rates between 8 and 4 kbits/s with a frame-length between 1.5 and 3 ms. Then we investigate the error sensitivity of these codecs, and describe two methods which were used to improve this error sensitivity. In Section 4 we describe a conventional forward adaptive ACELP codec operating at 4.7 and 6.5 kbits/s with a frame-length of 30 ms, and at 7.1 kbits/s with a frame-length of 20 ms. Finally in Section 5 the relative error resilience of these codecs is examined.

2. THREE LOW DELAY CODECS OPERATING BETWEEN 8 AND 4 KBITS/S

In this Section we describe three low delay CELP codecs, Codecs A, B, and C based loosely around the G728 16 kbits/s codec, whose attributes are summarised in Table 1 along with those of Codec D to be introduced at a later stage. The G.728 codec [1, 2] uses a frame-length of 5 samples or 0.625 ms, with 10 bits being used to code each 5 sample frame, giving a bit rate of 16 kbits/s. Backward adaption is used to derive the short term filter coefficients at both the encoder and the decoder, and hence no bits need to be transmitted to specify the filter coefficients used. Therefore all 10 bits per 5 sample frame are used to encode the filter excitation, which is vector quantized with a 7 bit

0-7803-3692-5/96 © 1996 IEEE

	Synthesis Filter	Long Term Predictor	Excitation Quantization	Frame Length	Bit Rate
Codec A	Backward Adapted p=50	None	8 Bit Shape plus 4 Bit Scalar Gain	1.5-3 ms	8-4 kbits/s
Codec B	Backward Adapted p=20	3 Tap Backward Adapted	8 Bit Shape plus 4 Bit Scalar Gain	1.5-3 ms	8-4 kbits/s
Codec C	Backward Adapted p=20	Partially Forward Adapted	8 Bit Shape plus 4 Bit Vector Gain	1.5-3 ms	8-4 kbits/s
Codec D	Forward Adapted p=10	Entirely Forward Adapted	12 Bit ACELP Shape plus 5+3 Bit Scalar Gain	20-30 ms	7.1-4.7 kbits/s

Table 1: Summary of Different Codecs Used

shape codebook and a 3 bit gain codebook. For each frame the best excitation is chosen using an Analysis-by-Synthesis (AbS) search.

We previously showed [5] how the G.728 codec could be modified to give a variable rate codec between 8 and 16 kbits/s with a graceful degradation in the speech quality of the codec, as the bit rate is reduced. Here we extend this work to produce three low delay codecs operating between 8 and 4 kbits/s. All three codecs use backward adaption (BA) for their synthesis filters, and transmit 12 bits per frame to represent the excitation to this filter. The codecs vary their bit rates by increasing the number of speech samples coded per frame from 12 to 24 samples, giving bit rates between 8 and 4 kbits/s. The difference between the three codecs lies in their use of Long Term Prediction (LTP). The first codec, referred to as Codec A, follows the philosophy of G728 and does not use LTP. Instead a very high order short term predictor is used. In both G728 and our Codec A the filter order $p = 50$ is used. An 8 bit shape and 4 bit gain codebook are used to represent this filter's excitation, giving a total of 12 bits per frame.

The second codec, referred to as Codec B, uses a short term filter of order $p = 20$, but also uses backward adapted long term prediction. The LTP delay, which takes integer values between 20 and 147, and the gain are determined at both the encoder and the decoder based on the correlations in the previous values of the synthesis filter's excitation. As in Codec A, an 8 bit shape and 4 bit gain codebook are used to vector quantize this excitation.

We also examined the effect of using forward adaption (FA) of the LTP in our low delay codec. As 7 bits would be needed to represent the LTP delay if this were forward adapted, and a total of only 12 bits are available to represent the filter excitation, we considered it impractical to use forward adaption for the LTP delay of our low bit rate, low delay, codecs. However, it is possible to use forward adaption of the LTP gain, and we implemented this in our third codec, referred to as Codec C. This codec, like Codec B, uses a short term filter of order $p = 20$ and backward adaption of the LTP delay. However the LTP gain is jointly determined with the fixed excitation gain in the AbS search of the excitation codebooks. The two gains are vector quantized using 4 bits, and again, 8 bits are used to vector quantize the excitation shape.

For all three codecs, the entries of the shape and gain

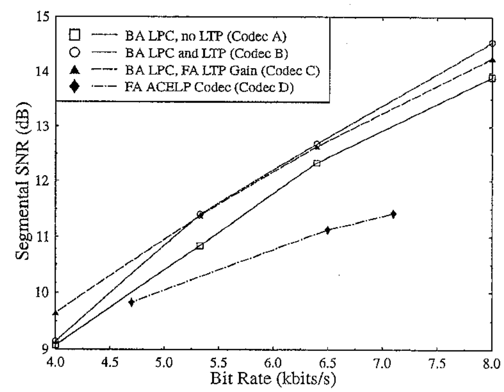

Figure 1: Segmental SNR versus bitrate performance of various CELP codecs

codebooks were trained using a closed-loop training technique similar to that described in [6]. We found that this training gave a significant improvement in all the codecs' performances. The segmental SNRs of the three codecs with frame-lengths of 12, 15, 18 and 24 and so bit rates of 8, 6.4, 5.3 and 4 kbits/s are shown in Figure 1. Also shown in this figure is the performance of the forward adapted Algebraic CELP (ACELP) codec to be described in Section 4. It can be seen by comparing Codec A and Codec B in Figure 1 that the addition of backward adapted LTP increases the segmental SNR of the codec by over 0.5 dB at 8 kbits/s, but that the effectiveness of entirely backward adapted LTP decreases, as the bit rate decreases so that it gives almost no gain at 4 kbits/s. Similarly, it is clear that although Codec B outperforms Codec C at 8 kbits/s, as the bit rate is reduced, the codec using entirely backward adaptive LTP is more seriously affected so that at 4 kbits/s Codec C significantly outperforms Codec B. In informal listening tests we found that all three codecs gave close to toll quality speech at 8 kbits/s, but at 4 kbits/s Codec C using forward adaption for the LTP gain sounded clearly better than Codec A or B and gave speech of communications quality.

3. THE ERROR SENSITIVITY OF THE 6.4 KBITS/S LOW DELAY CODECS

In this Section we consider the error sensitivity of our low delay codecs. For the sake of convenient comparison, the error sensitivities of the codecs operating with a frame-length of 15 samples and a bit rate of 6.4 kbits/s are detailed in this Section, and compared to the forward adaptive 6.5 kbits/s codec in Section 5. However, similar conclusions apply at the other bit rates.

It is well known that codecs using backward adaption for both the LTP delay and gain are very sensitive to bit errors, and this is why LTP was not used in G728 [1]. Thus, as expected, we found that Codec B gave a very poor performance, when subject to even a relatively low Bit Error Rate (BER). Unfortunately, we also found similar results for Codec C which, although it used backward adaption for the LTP delay, used forward adaption for the LTP gain. We therefore decided that neither Codec B nor Codec C were suitable for use over noisy channels, and examined the error sensitivity of Codec A, which does not use LTP. At 6.4 kbits/s this codec transmits only 12 bits per 15 sample frame from the encoder to the decoder. Of these 12 bits 8 are used to represent the index of the codebook entry used from the shape codebook, and the remaining 4 bits are used to represent the index of the gain codebook entry used. The error resilience of these bits can be significantly improved by careful assignment of codebook indices to the various codebook entries. Ideally, each codebook entry would be assigned an index so that corruption of any of the bits representing this index will result in another entry being selected in the decoder's codebook which is in some way "close" to the intended codebook entry. If this ideal can be achieved, then the effects of errors in the bits representing the codebook indices will be minimised.

Consider first the 8 bit shape codebook. Initially the 256 available codebook indices are effectively randomly distributed amongst the codebook entries. We seek to rearrange these codebook indices so that when the index representing a codebook entry is corrupted, the new index will represent a codebook entry that is "close" to the original entry. In our work we chose to measure this "closeness" by the squared error between the original and the corrupted codebook entries. We considered only the effects of single bit errors among the 8 codebook bits because at reasonable Bit Error Rates (BERs) the probability of two or more errors occurring in 8 bits will be small. Thus for each codebook entry the "closeness" produced by a certain arrangement of codebook entries is given by the sum of the squared errors between the original codebook entry and the eight corrupted entries that would be produced by inverting each of the 8 bits representing the entry's index. The overall "cost" of a given arrangement of codebook indices is then given by the closeness for each codebook entry, weighted by the probability of that codebook entry being used. Thus the cost we seek to minimise is given by

$$\text{Cost} = \sum_{j=0}^{255} P(j) \left[\sum_{i=1}^{8} \left(\sum_{n=1}^{15} (c_j(n) - c_j^i(n))^2 \right) \right] \tag{1}$$

where $P(j)$ is the probability of the j'th codebook entry being used, $c_j(n)$, $n = 1 \cdots 15$, is the j'th codebook entry and $c_j^i(n)$ is the entry that will be received if the index j is transmitted but the i'th bit of this index is corrupted.

The problem of choosing the best arrangement of the 256 codebook indices among the codebook entries is similar to the well-known travelling salesman problem. In this problem the salesman must visit each of N cities, and must choose the order in which he visits the cities so as to minimise the total distance he travels. As N becomes large it becomes impractical to solve this problem using an exhaustive search of all possible orders in which he could visit the cities - the complexity of such a search is proportional to $N!$ Instead, a non-exhaustive search must be used which we hope will find the best order possible in which to visit the N cities.

The minimisation method of simulated annealing has been successfully applied to this problem [7], and has also been used by other researchers as a method of improving the error resilience of quantizers [8]. The procedure operates as follows. The system starts in an initial state, which in our situation is an initial assignment of the 256 codebook indices to the codebook entries. A temperature like variable T is defined, and possible changes to the state of the system are randomly generated. For each possible change the difference $\Delta Cost$ in the cost between the present state and the possible new state is evaluated. If this is negative, ie the new state has a lower cost than the old state, then the system always moves to the new state. If on the other hand $\Delta Cost$ is positive then the new state has a higher cost than the old state, but the system may still change to this new state. The probability of this happening is given by the Boltzmann distribution

$$\text{prob} = \exp\left(\frac{-\Delta Cost}{kT}\right) \tag{2}$$

where k is a constant. The initial temperature is set so that kT is much larger than any $\Delta Cost$ that is likely to be encountered, so that initially most offered moves will be taken. As the optimization proceeds, the 'temperature' T is slowly decreased, and the number of moves to states with higher costs reduces. Eventually kT becomes so small that no moves with positive $\Delta Cost$ are taken, and the system comes to equilibrium in what is hopefully the global minimum of its cost.

The advantage of simulated annealing over other optimization methods is that it is not likely to be deceived by local minima, approaching the global minimum of the function to be minimised. In order to make this likely to happen, it is important to ensure that the temperature T starts at a high enough value, and is reduced suitably slowly. We followed the suggestions in [7] and reduced T by 10% after every $100N$ offered moves, or every $10N$ accepted moves, where $N = 256$ is the number of codebook entries. The initial temperature was set so that kT was equal to ten times the highest value of $\Delta Cost$ that was initially encountered. The random changes in the state of the system were generated by randomly choosing two codebook entries and swapping the indices of these two entries.

The effectiveness of the simulated annealing method in reducing the cost given in Equation 1 is shown in Figure 2. This graph shows the cost of the present arrangement of codebook indices against the number of arrangements of codebook indices which have been attempted by the minimisation process. The initial randomly assigned arrangement of indices to codebook entries gives a cost of 1915. As can be seen in Figure 2, initially the temperature T is high and so many index assignments, which have a higher cost than this

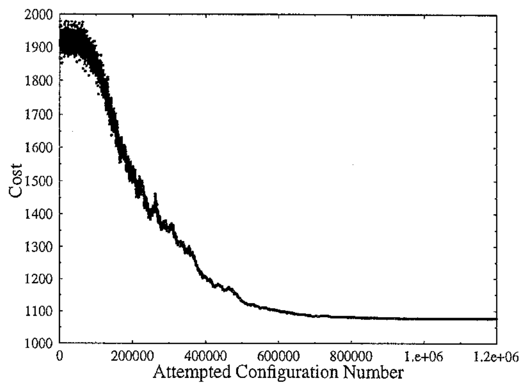

Figure 2: Reduction in Cost Using Simulated Annealing

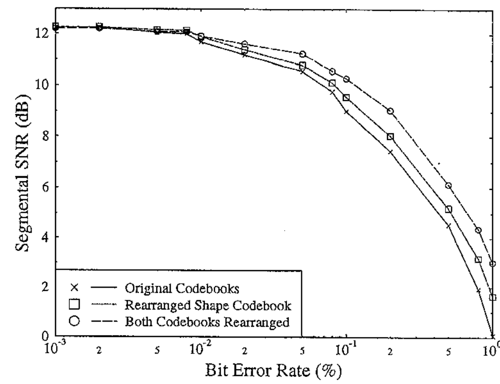

Figure 3: The Error Sensitivity of Our Low Delay 6.4 kbits/s Codec

are accepted. However, slowly as the number of attempted configurations increases the temperature T decreases, and so fewer re-arrangements which increase the cost of the present arrangement are accepted. Thus, as can be seen in Figure 2, the cost of consecutive arrangements slowly falls, and the curve tapers, as the temperature increases, thereby accepting less re-arrangements, which increase the cost of the present arrangement are accepted. The cost of the final assignment of codebook indices to codebook entries is 1077, which is nearly half of the original cost.

The effectiveness of this re-arrangement of codebook indices in increasing the resilience of the codec to errors in the bit stream between its encoder and decoder can be seen in Figure 3. This graph shows the segmental SNR variation of our 6.4 kbits/s low delay Codec A versus BER. The solid line shows the performance of the codec with the original codebook index assignment, and the lower dashed line shows the performance when the shape codebook indices are re-arranged as described above. It can be seen that at BERs of between 0.1% and 1% the codec with the re-arranged codebook indices has a segmental SNR about 0.5 to 1 dB higher than the original codec.

Apart from the 8 shape codebook bits, which Codec A transmits from its encoder to the decoder, the only other information that is explicitly transmitted are the 4 bits representing the gain codebook entry selected. Initially indices were assigned to the 16 gain codebook entries using the simple Natural Binary Code (NBC). However, because the gain codebook levels do not have an equiprobable distribution, this simple assignment can be improved upon in a similar way to that described for the shape codebook above. Again, we defined a cost function that was to be minimised. This cost function was similar to that given in Equation 1, except because the gain codebook is scalar, whereas the shape codebook has a vector dimension of 15, no summation over n is needed in the cost function for the gain codebook index arrangement. We used simulated annealing again to reduce the cost function over that given using a NBC, and found that we were able to reduce the cost by over 60%. The effect of this re-arrangement of the gain codebook indices is shown by the upper curve in Figure 3, which gives the performance of Codec A with both the gain and shape codebooks re-arranged. It can be seen that the re-arrangement of the gain codebook indices gives a further improvement in the error resilience of the codec, and that the codec with both the shape and gain codebooks re-arranged has a segmental SNR more than 1 dB higher than the original codec at BERs around 0.1%.

4. A FORWARD ADAPTIVE ACELP CODEC

The final codec in our comparison was a forward adaptive ACELP codec, which we refer to as Codec D. This codec, which is described in detail in [9], operates at 6.5 and 4.7 kbits/s with a 30 ms frame-length and at 7.1 kbits/s with a 20 ms frame-length. In each frame 34 bits are used to quantize the forward adapted LPC coefficients. Each frame is split into either 4 or 6 sub-frames, depending on the bit rate, and for each sub-frame 12 bits are used to represent the algebraic codebook entry selected, 7 bits are used to represent the forward adapted LTP delay, and a total of 8 bits are used to scalar quantize the LTP and ACELP gains. This gives a total of either 196 or 142 bits per 20 or 30 ms frame, giving bit rates of 4.7, 6.5 and 7.1 kbits/s.

The segmental SNR of this FA codec is shown in Figure 1. It can be seen that it gives a significantly lower segmental SNR than the low delay codecs, but that this difference decreases as the bit rate is reduced. Furthermore, forward adaption of the synthesis filter parameters gives an improvement in the spectral match that can be achieved between the original and the reconstructed speech. This improvement in the spectral match is not adequately reflected in the segmental SNR. Thus although, as indicated by Figure 1, the backward adaptive codecs do give better reconstructed speech quality than Codec D at high rates, at lower rates Codec D provides the best speech quality.

The error sensitivity of Codec D was improved using the techniques described in [9, 10]. These techniques include a way to correct 25% of all bit errors that occur in the 34 bits representing the LPC coefficients of the codec, and smoothing of the ACELP gains to correct errors that occur in these bits. The error sensitivity of the resulting codec is detailed in the next section, where it is compared to that of

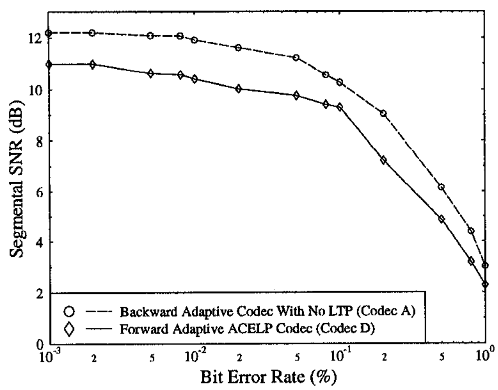

Figure 4: A Comparison of the Bit Error Sensitivities Codecs A and D.

the backward adaptive Codec A.

5. RELATIVE ERROR SENSITIVITIES OF THE FORWARD AND BACKWARD ADAPTIVE CODECS

As noted in Section 3, both Codec B and Codec C, which use long term prediction and employ backward adaption of the LTP delay, are extremely sensitive to channel errors. However Codec A is much more robust, and its segmental SNR at various BERs is compared in Figure 4 to that of the forward adaptive Codec D. Codec A uses the re-arranged shape and gain codebooks as described in Section 3, and Codec D uses the methods of improving error sensitivity mentioned above. It can be seen that surprisingly Codec A is almost as robust to channel errors as the forward adaptive Codec D. Both codecs give a graceful degradation in their reconstructed speech quality at BERs up to about 0.1%, but provide poor reconstructed speech quality for BERs much above this.

6. CONCLUSIONS

In this paper we have detailed three backward adaptive, low delay CELP codecs operating between 8 and 4 kbits/s. Both the reconstructed speech quality and the error sensitivity of these codecs were compared to that of a much higher delay forward adaptive ACELP codec. We found that at high bit rates the backward adaptive codecs provide superior speech quality, but as the bit rate is reduced towards 4 kbits/s, the forward adaptive ACELP codec provides better speech quality. Furthermore, we investigated and compared the error sensitivity of the codecs, and found that error resilience similar to that of forward adaptive codecs can be achieved with backward adaptive codecs.

7. ACKNOWLEDGEMENT

The financial support of the EPSRC, UK in the framework of the contract GR/K74043 is gratefully acknowledged.

8. REFERENCES

[1] Juin-Hwey Chen, Richard V. Cox, Yen-Chun Lin, Nikil Jayant and Melvin J. Melchner, "A Low-Delay CELP Coder for the CCITT 16 kb/s Speech Coding Standard," *IEEE Journal on Selected Areas in Communications*, pp. 830–849, June 1991.

[2] "Coding of Speech at 16 kbit/s Using Low-Delay Code Excited Linear Prediction." CCITT Recommendation G.728, 1992.

[3] Redwan Salami, Claude Laflamme, Jean-Pierre Adoul and Dominique Massaloux, "A Toll Quality 8 Kb/s Speech Codec for the Personal Communications System (PCS)," *IEEE Transactions on Vehicular Technology*, pp. 808–816, August 1994.

[4] "Coding of Speech at 8 kbit/s Using Conjugate-Structue Algebraic Code-Excited Linear Prediction (CS-ACELP)." ITU Draft Recommendation G.729, February 1996.

[5] J.P. Woodard, J.M. Torrance and L. Hanzo, "A Low Delay Multimode Speech Terminal," *Proceeding of the IEEE Conference on Vehicular Technology*, pp. 213–217, April 28 - May 1 1996.

[6] Juin-Hwey Chen, "High-Quality 16 Kb/s Speech Coding with a One-Way Delay Less Than 2ms," *Proceedings of the IEEE Conference on Acoustics, Speech and Signal Processing*, pp. 453–456, 1990.

[7] William H. Press, Saul A. Teukolsky, William T. Vetterling and Brian P. Flannery, *Numerical Recipes in C.* Cambridge University Press, 1992.

[8] Sonia L.Q. Dall'Agnol, J. Roberto B. De Marca and Abraham Alcaim, "On the Use of Simulated Annealing for Error Protection of CELP Coders Employing LSF Vector Quantizers," *Proceeding of the IEEE Conference on Vehicular Technology*, vol. 3, pp. 1699–1703, June 1994.

[9] Jason Paul Woodard, *Digital Coding of Speech Using Code Excited Linear Prediction.* PhD thesis, University of Southampton, 1995.

[10] L. Hanzo and J.P. Woodard, "An Intelligent Multimode Voice Communications System for Indoor Communications," *IEEE Trans. on Vehicular Technology*, pp. 735–748, Nov 1995.