NISTIR 5503

Channel Coding for Code Excited Linear Prediction (CELP) Encoded Speech in Mobile Radio Applications

Ehud Bracha
Nariman Farvardin
Yaacov Yesha

U.S. DEPARTMENT OF COMMERCE
Technology Administration
National Institute of Standards
and Technology
Advanced Systems Division
Digital Signal Processing Group
Gaithersburg, MD 20899

NIST

NISTIR 5503

Channel Coding for Code Excited Linear Prediction (CELP) Encoded Speech in Mobile Radio Applications

Ehud Bracha
Nariman Farvardin
Yaacov Yesha

U.S. DEPARTMENT OF COMMERCE
Technology Administration
National Institute of Standards
and Technology
Advanced Systems Division
Digital Signal Processing Group
Gaithersburg, MD 20899

August 1994

U.S. DEPARTMENT OF COMMERCE
Ronald H. Brown, Secretary

TECHNOLOGY ADMINISTRATION
Mary L. Good, Under Secretary for Technology

NATIONAL INSTITUTE OF STANDARDS
AND TECHNOLOGY
Arati Prabhakar, Director

CHANNEL CODING FOR CODE EXCITED LINEAR PREDICTION (CELP) ENCODED SPEECH IN MOBILE RADIO APPLICATIONS

Ehud Bracha[1], Nariman Farvardin, and Yaacov Yesha

1. Introduction

The Digital Signal Processing Group at NIST is involved in an effort aimed at the development of a National Wireless Performance Benchmarking Program. This program seeks to provide tools, performance metrics, methodologies and testbed facilities to the wireless industry and users, so as to allow consistent and impartial performance measurement of wireless communication systems. See [1] for details.

The systems involved may include speech, image and general data communication systems. Due to the limited bandwidth of wireless channels, some form of data compression must be used in order to accommodate more users in a given bandwidth, and different compression algorithms are needed for each type of data. For example, the Code Excited Linear Prediction (CELP) algorithm has been adopted by the Federal Government as Federal Standard (FS) 1016, for use in voice coding [2]. Wireless channels generally introduce higher error rates than do wireline channels. Therefore, channel coding is also needed, in order to allow for error detection and correction. It is assumed that the reader has a working knowledge of CELP and is familiar with the mobile radio environment.

As part of the benchmarking program at NIST, research is currently being undertaken in which the mobile radio channel effects on CELP encoded speech data are studied. This research uses a mix of software and hardware tools to test the channel encoding algorithms and evaluate their error correction performance; see [3]-[5] for a description of previous work. The approach selected for the current study is based on software simulations on a SUN[2] workstation. At later stages of the research Digital Signal Processing (DSP) hardware should be added in order to increase simulation speeds and to allow real-time simulations.

---

[1] On Leave from Rafael, Israel

[2] Certain commercial equipment, instruments, or materials are identified in this paper in order to specify the experimental procedure adequately. Such identification is not intended to imply recommendation or endorsement by the National Institute of Standards and Technology, nor is it intended to imply that the materials or equipment identified are necessarily the best available for the purpose.

The selection of speech data and CELP for this study was motivated by their relative simplicity, by the fact that the vast majority of data transmitted today is speech data, and by the readily available simulation software of the well-established algorithms. At later stages of the research, issues related to image data and general data transmission over wireless communication channels would certainly have to be addressed as well.

Examples of related work on channel coding for speech data over wireless channels can be found in [6]-[8]. These use different error correction codes to overcome channel errors characteristic of mobile radio channels. The current study is aimed at gaining insight into the problems associated with wireless channels, thus allowing a better design of channel coding systems and the benchmarking paradigm, in general.

This report is concerned with a software simulation of channel coding of CELP encoded speech data on wireless communication systems. Since errors in different CELP bits have different effects over the reconstructed speech, the report first describes the tests conducted to identify CELP bit sensitivity to random errors. Objective measurement of speech quality was done using Segmental Signal-to-Noise Ratios (SEGSNR) for the purpose of comparison with other studies, such as [8]. Secondly, bit protection tests are described, in which simulated radio channel errors were used to identify the most important CELP bits by measuring speech quality, both objectively and subjectively. Once the important CELP bits were identified, error correction codes were used to protect these selected bits, so as to improve the output speech quality under simulated channel errors. Three error correction codes were used, all using Reed-Solomon (RS) codes, and all protected 84 CELP bits. The first code was suggested by [8] and the other two were developed at NIST. Both NIST codes are shown to be better than the first code, in protecting CELP bits under simulated channel errors. It is also shown that a variation of the first NIST code gives the best performance among the two NIST codes.

Section 2 provides a detailed description of the tests performed in this study and explains how SEGSNR was computed and how channel errors were simulated. Section 3 describes the bit error sensitivity tests and their results. Section 4 describes the bit protection sensitivity tests and their results. Section 5 describes the process used to select the CELP bits to be protected and the three error correction codes compared in this study. Section 6 contains the results of the comparison between the error correction codes, and, finally, Section 7 includes some concluding remarks about channel coding for wireless communication systems.

2. Detailed Description of Case Study

The software simulation system used in the current study included the following components of a wireless communication system, as shown in Figure 1 below:

1. Source encoding;
2. Channel encoding;
3. Radio channel (including access method, modulation and demodulation, noise and multipath fading, and receiver simulations);
4. Channel decoding;
5. Source decoding.

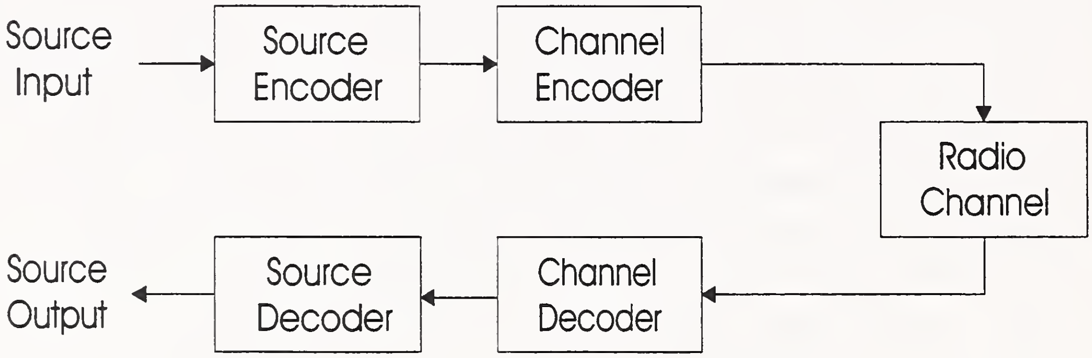

Figure 1: Communication System Simulation Architecture

At the early stages of the study, only the source encoder and decoder were used with the radio channel simulation, in order to study the relative importance of the different bits in each CELP frame. The channel encoder and decoder simulations were added later, to test the effect of error correction codes over wireless channels.

The source encoder included the CELP analysis stage, which compressed the original speech data from 8000 samples per second of 16 bits per sample (128,000 bps), to 4800 bps, or 144 bits per CELP frame each 30 msec [2]. The channel encoder operated on one CELP frame at a time, and used RS error correction codes for groups of bits in each CELP frame, encoding each group with its own codeword size and error correction capability. The resulting bit rate was 7200 bps, as used in [8]. The radio channel simulation was performed in an off-line fashion, as described in [3]. The resulting bit error masks were used to mask the corresponding data bits transmitted over the simulated radio channel. In the first tests in which CELP data was subjected to channel errors, 4800 bit error masks per second were used. In the last test in which the RS codes were used, 7200 bit error masks per second were used.

In order to minimize the effects of burst errors characteristic of multipath fading channels, interleaving of symbols was used in the last tests, so that symbols in the same codewords were separated as much as possible. The interleaving was performed as part of the error masking process.

The channel decoder included a RS decoder for each codeword, which decoded and corrected errors in the channel data. The resulting data was again CELP frames of 4800 bps, possibly corrupted by noise. Finally, the source decoder included the CELP synthesis stage, which reconstructed the output speech data to the same rate as the input data. The synthesizer was operated in two modes. In both of the bit sensitivity tests the smoothing was disabled in order to emphasize the importance of the CELP bits. In the error correction test, the smoothing was activated so as to achieve the best improvement in quality, both objectively and subjectively. Since fixing of the Line Spectral Parameters (LSPs) in CELP is independent of the smoothing of other parameters, it was active in all the tests.

The rest of this section describes the tests performed in this study, explains how SEGSNR was computed and provides some information on the radio channel simulations used.

2.1 Detailed Description of Tests

The first set of tests consisted of the bit error sensitivity tests. The tests were intended to directly identify which CELP bits are more important by subjecting single bits in randomly selected CELP frames to errors. One test was run for each of the 144 CELP bits, while the frames in error in each one of the 144 tests were exactly the same. The purpose of the random selection of the frames was to provide a better statistical validity of the tests. The measurement used was SEGSNR of the output speech file relative to the input file. For this test, an error mask file with random 0's and 1's was generated, where a 1 meant an error in the corresponding CELP frame. A special masking program was used, which read the bit number to be masked. The program read in each CELP frame, read the next mask value from the random mask file, and if the value was a 1, it inverted the desired bit in the CELP frame, while the rest of the bits were left unchanged. The resulting output file then contained random errors in the selected CELP bit position. For each such output file SEGSNR was computed, and after 144 runs there were 144 SEGSNR values which were plotted on a graph as a function of bit number. On this graph it is easy to observe which bits are the most sensitive to errors by simply looking for those with the largest decrease in SEGSNR. Section 3 below provides more details on the tests performed and results from the bit error sensitivity tests.

The tests described above provided a direct coupling between CELP bits and their effect on the audio output. However, the Bit-Error-Rate (BER) introduced in such tests was relatively low and did not represent actual wireless channel error rates. For example, even when the same single bit is corrupted in every frame, the resulting error rate is only 1 out of 144, namely, about 0.7%. In contrast, the error rates present on wireless channels may be as high as 10%, and the errors are usually bursty in nature, i.e., they span more than one bit. Hence, another testing approach was needed.

Since the general approach in this study is based on protection of important CELP bits with the channel encoder, the second set of tests included bit protection sensitivity tests. In these tests, an 8% BER simulated radio channel bit error mask was used, so that actual channel conditions, including burst errors, were tested and the errors were no longer limited to single bits. The CELP encoded speech samples used in these tests were the same as those used in the previous set of tests. In order to identify which bits are important, a masking program which allowed the protection of a set of bits in each CELP frame was used, with the rest of the bits still subjected to simulated channel errors. In order to identify the contribution of single bits, a set of tests was initially performed such that in each test only a single bit was protected and the resulting SEGSNR was computed. After 144 runs the 144 SEGSNR values were plotted on a graph as a function of bit number. On this graph it is easy to observe which bits are the most important by simply looking for those with the largest increase in SEGSNR, as compared to SEGSNR with no protection at all.

After the first group of important bits was identified, a second set of bit protection tests was run in which this first group of bits was protected, and then one more bit from the remaining bits was also protected for each test and SEGSNR was measured. The relative importance of the remaining bits was again found by plotting the relative improvement of SEGSNR gained by protecting single bits. This procedure was repeated as long as further protection of bits produced a noticeable improvement in SEGSNR. The relative importance of the CELP bits was found with this procedure simply by the order of the tests, so that the bits protected in each test were selected based on the results of the previous test. Section 4 below provides more details on the tests performed and results of the bit protection sensitivity tests.

The last set of tests was intended to compare three error correction codes under simulated channel errors, for use in the channel encoder and decoder, both objectively and subjectively. The codeword structure and bits for the first code were taken from [8]. For the two NIST codes, the results of the bit error and bit protection sensitivity test were used to produce a list of CELP bits in order of importance, and the bits to be protected by the error correction code were selected from this list. These bits were grouped into codewords, to be encoded by the tested channel encoder, masked (and interleaved) by a masking program, and then decoded by the tested channel decoder and CELP synthesized into the reconstructed speech samples. Each code was tested with four different channel bit error mask files, each file representing different radio channel conditions, and a SEGSNR was computed for each test. Comparison of the three systems was then done by comparing the resulting SEGSNRs for the same bit error mask file and by listening to the actual speech outputs. See Sections 5 and 6 for more details.

2.2 Measurements of Quality

Objective measurement of speech quality was made by using SEGSNR of the output file relative to the original input file. This measure of quality was used in all the tests described above. SEGSNRs were computed for a 300 second long audio file of 16-bit PCM samples, which translates into 10000 CELP frames. The large number of frames is needed to increase the statistical validity of the results. The segment length used in this study was 240 samples, or 1 CELP frame of 30 msec, at 8000 samples/sec. The signal energy was taken from the original frame of samples and the noise energy was computed as the energy of the difference between the CELP synthesized output samples and the original frame samples. Each segment's SNR was then computed using the formula:

$$SNR_{Segment} = 10\log\left(\frac{Energy_{Signal}}{Energy_{Noise}}\right).$$

The overall SEGSNR was simply the average of all the segment SNRs, with segments having zero signal or zero noise excluded from this average.

Informal listening tests were also used to subjectively verify the objective test results. These were done so that the listeners did not know in advance what was being played, and they had to rate the relative quality of pairs of output files, using criteria such as amount of blasts, intelligibility and noise present in the output samples.

The audio file selected for this study was taken from the TIMIT database from NIST ([9]), which is publicly available. The 300 second long file was built from shorter segments of speech.

2.3 Radio Channel Simulations

The bit protection tests and the tests with the error correction codes used bit error masks to simulate the effects of errors on the radio channel. The masks represented the overall effect of the channel on the data bits, including the access method, the modulator and demodulator, and the multipath fading and noise characteristics of mobile radio channels. Once the bit error masks were generated, the simulation was simply done by using an Exclusive-OR operation between the data bits and the bit error mask bits.

The masks were generated by two radio channel simulation programs, one developed by AT&T

and Hughes Network Systems [10], and one received from NSA [11]. The AT&T/Hughes simulator simulates a half-rate TDMA channel, QDPSK modulation, Gaussian noise and multipath fading controlled by two input parameters: receiver speed and channel SNR. Receiver speed refers to the rate at which the receiver moves in space, measured in miles per hour (MPH). Each CELP voice channel uses 6500 bits per second, where 4800 bits are used for CELP data and the remaining 1700 bits are not used. The output of this simulator includes a bit mask for each simulated data bit, showing if there is a hard error in the data bit, and a soft decision value representing the receiver's confidence in the value received. These two outputs are grouped into an integer, and extracting the hard errors bit masks requires further processing.

The parameters for the NSA simulator are as follows: receiver speed, channel SNR, radio frequency and transmission bit rate. This allows simulation of the 7200 bps needed for the RS coded data. This simulator simulates QDPSK modulation, Rayleigh multipath fading, and Gaussian noise as well. The output of this simulator is also a file of bit masks and an optional file of soft decision values, so further processing of the bit masks is not needed.

The need to adjust the bit rate in the error correction code tests motivated the change to the NSA simulator, since this simulator provides a higher flexibility in the definition of the bit rates. Comparison between the simulators with similar parameter values produced similar results.

3. Bit Error Sensitivity Tests

3.1 Purpose of Tests

These tests provide information that may be used to determine the relative importance of the various CELP bits, by directly identifying the effect of each bit on the resulting output audio.

3.2 Method of Testing

Suppose that one would like to answer the following question: If all the bits except for one bit could be fully protected against channel errors, which bit should remain unprotected? In order to find out the answer, one could subject the CELP bits to channel errors, while fully protecting all the bits except one. This can be done for all the 144 bits, one bit at a time and the SEGSNR can be computed for each bit. Then, the bit that should be left unprotected is the one for which the SEGSNR is highest. Next, suppose that all the 144 bits except for k bits can be protected. One could use the following heuristic method: the k bits with the highest SEGSNR will be left unprotected. In other words, the (144-k) bits with the lowest SEGSNR will be protected. The same heuristic could be used to determine which bits should be protected by error control coding, which is of course very different from full protection. The actual SEGSNR values could be used also to determine relative sensitivities of bits to channel errors.

In the present work, a simplified heuristic was used. This heuristic corresponded to a channel bit error mask which was all 1's (i.e. all errors) in certain frames, and all 0's (i.e. no errors) in all other frames, where the selection of frames with errors was done by using a random number generator. In this case, protecting all the bits except one is the same as introducing an error in the unprotected bit in the selected frames.

The frame error rates tested included 1%, 2%, 5%, 10% and 50%. The most significant result was obtained from the 50% frame error rate test, since half of the bits in each test were corrupted, and the effect of the bit in error was noticeable. Note that testing with a 100% frame error rate is possible, but this removes the effects of randomness from the test. SEGSNR values were computed for all 144 bits.

3.3 Summary of Results

Figure 2 shows a graph of 145 SEGSNRs as a function of CELP bit number in unpermuted order. These SEGSNRs resulted from 50% random frame errors in each one of the 144 CELP bits, where bits are numbered from 1 to 144. Bit 0 refers to SEGSNR without errors, which is 3.83 dB. The "future bit" (139), the Hamming error control bits (140-143) and the Sync bit (144) are included for the sake of completeness. These bits are not used in the CELP synthesis, so they cause no degradation in SEGSNR relative to SEGSNR without errors.

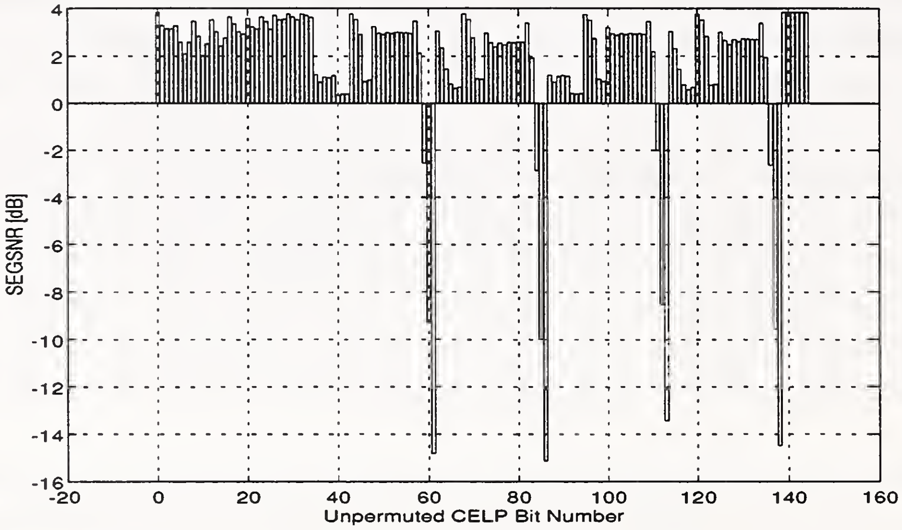

Figure 2: SEGSNR at 50% Frame Error Rate

3.4 Initial Conclusions from Bit Error Tests

The initial conclusions from Figure 2 regarding CELP bit sensitivity to random errors are that bits 59-61, 84-86, 111-113, and 136-138 are highly sensitive. These are the 3 most significant bits of the codebook gain (CBGAIN) of each one of the 4 subframes in each CELP frame. The next group of bits showing high sensitivity to errors are bits 40-42, 65-67, 92-94 and 117-119, which are the 3 most significant bits of the pitch delay and delta delay parameters. These bits are closely followed by bits 46-47, 71-72, 98-99, and 123-124, which are the 2 most significant bits of the pitch gain parameters.

The bits least sensitive to errors are the codebook index parameters (bits 48-56, 73-81, 100-108 and 125-133). These bits show some degradation in SEGSNR, but it is almost the same for each codebook index group of bits. The LSP bits (1-34) show also relatively small changes in SEGSNR, with the more significant bits showing a larger decrease in SEGSNR, as expected. For example, for LSP1 (bits 1-3) bit 3 has the lowest SEGSNR, and for LSP2 (bits 4-7) bit 6 shows the largest decrease, and so on. The fact that the most significant bit in each LSP has a slightly better SEGSNR than the bit next to it may be attributed to the LSP fixing, which tends to reduce the overall error in the LSPs, and especially in their most significant bits.

In order to check for a match between the objective tests above to subjective test results, output speech files for selected important bits were saved and listening to them produced the following results. For bit 61 (CBGAIN) the resulting audio indeed produced very large distortion in volume (blasts). For bit 42 (pitch delay) a noticeable distortion in pitch was observed. For bit 47 (delta delay) the distortion was minimal. These results match the relative importance of the bits as found in the objective tests.

4. Bit Protection Sensitivity Tests

4.1 Purpose of Tests

The purpose of the bit protection sensitivity tests is to determine the relative importance of protecting various bits against channel errors. Error protection schemes generally require data rates higher than that of the original data because of additional error control bits. The available bandwidth imposes a limit on the number of bits available for error control. It is plausible that in order to efficiently protect the original signal, bits that are more sensitive to channel errors should receive stronger error protection. This generally requires use of more bandwidth for those protected bits, resulting from use of lower rate error control codes. Low sensitivity to error and limited bandwidth may result in deciding to provide no error protection for some bits. Following this strategy, the intention is to use RS codes of various rates to provide unequal protection based on bit sensitivity to errors.

4.2 Method of Testing

The tests included measuring the SEGSNR for speech that was synthesized by CELP from CELP coded speech that was subjected to channel errors. Then, the SEGSNR was measured when a single bit was fully protected, while no other bit was protected. This was done separately for every bit. The CELP synthesizer used in these tests had the Hamming error control coding and the parameter smoothing turned off, because they are forms of error protection and the intention was to protect only one bit at a time. High SEGSNR increases correspond to bits that are more sensitive to channel errors.

It is assumed that, as in the system suggested in [8], 84 bits will be protected by error control coding. Consider the bit whose protection results in the highest SEGSNR. It is believed that this bit should be among the protected bits. In other words, it is believed that there exists an error control scheme for 84 bits including that bit, that results in SEGSNR that is not lower than the SEGSNR of any error control scheme for 84 bits that excludes that bit. And if this is not the case, it seems likely that at least selecting this bit for protection is reasonable in the following sense: There exists an error control scheme for 84 bits including that bit that results in SEGSNR that is not substantially lower than the SEGSNR achieved by any error control scheme for 84 bits excluding that bit. This is a heuristic argument that can be generalized as follows. Suppose that it was determined that it is reasonable to include a certain group of k bits among the 84 protected bits. Then an additional bit to be protected is selected by protecting each one of the remaining 144-k bits, one at a time, while the above k bits are always protected in each one of these 144-k experiments. Among these 144-k bits, the bit whose protection results in the highest SEGSNR is added to the group of bits to be protected. This method is iterated until a group of 84 bits to be protected is determined. As mentioned above, this seems to be a good method for finding a group of 84 bits for full protection. Intuitively, it also seems to be a good method for finding a group of 84 bits for protection by error control coding, which is clearly far from full protection. Using a heuristic method is inevitable since it is infeasible to search through all combinations of 84 out of 144 bits and all possible RS error control schemes for each combination.

In addition to determining the 84 bits to be protected, it was also necessary to compare the sensitivities of those 84 bits, in order to determine the various error protection levels for those bits. The order of sensitivity is the order of adding bits to the group of protected bits. The bits that are added earlier to this group are presumed to have higher sensitivity.

In order to speed up the process, a variation on the above method was actually used. After each step of finding the SEGSNR for protected bits, sometimes more than one bit was added to the group of protected bits. The bits added were always those with the highest increase in SEGSNR. A particular justification to adding more than one bit exists when all the bits added are at the same position in the same parameter for different subframes. One such example is the most significant bits of the four codebook gains.

The tests used a channel bit error mask which simulated 8% BER, generated with the AT&T/Hughes simulator, with receiver speed of 3 MPH and channel SNR of 10 dB.

4.3 Summary of Results

Figure 3 shows the results of the first set of bit protection tests. At first, the analyzed CELP channel file was masked with no bit protection at all. The SEGSNR computed from the resulting output was -10.51 dB, as shown in Figure 3. Then, a set of tests were run, one test for each CELP bit being protected and using the same mask file. The resulting changes in SEGSNRs shown in Figure 3 represent the effect of each CELP bit being protected relative to the SEGSNR with no protection. Note that bits 139-144 were excluded from the tests, since they are control bits, which are not used in the CELP synthesis. This was done in order to save CPU time.

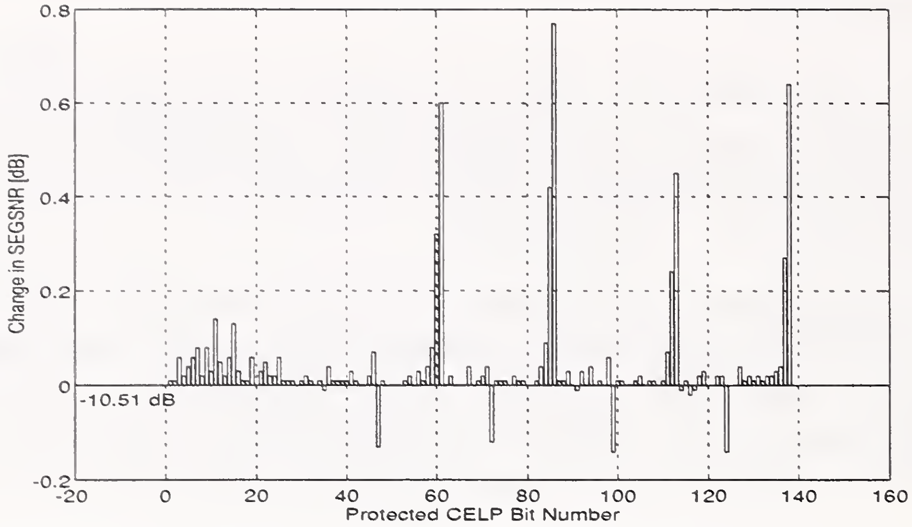

Figure 3: SEGSNR Improvement with Single CELP Bit Protection

The first group of highly sensitive CELP bits were selected from Figure 3 simply by looking for the bits producing the highest increase in SEGSNR, when protected. Clearly, these bits were 61, 86, 113 and 138, which are the most significant bits in each one of the CBGAIN parameters of the 4 CELP subframes. An interesting observation at this point was that bits 47, 72, 99 and 124 actually produced a decrease in SEGSNR, meaning that protecting each one of them in the presence of other errors was worse than leaving them with no protection. These bits are the most significant bits of the 4 pitch gain parameters in the CELP frames. This behavior is believed to be a result of the asymmetric coding of the pitch gain parameters. In addition, at the high bit error rate used, many other bits are also in error and then protecting these bits only, without protecting the other bits in the same parameters, does not improve the SEGSNR.

The statistical validity of the results above was verified by plotting the number of frames in which each bit was actually in error. This was done in order to insure that all bits had similar error rates overall. Figure 4 shows a plot of the number of times each CELP bit was in error, together with the minimum, mean, maximum and standard deviation of the error rates. Since the variation in the number of errors in the different bits is not significant, it is reasonable to assume that this variation does not contribute significantly to the changes in SEGSNR, but rather the real importance of the bit being tested affects the SEGSNR change.

Based on the results of the first set of protection tests, the second set of tests was designed. The results are shown in Figure 5. In the first test in this set, only bits 61, 86, 113 and 138 were protected. The SEGSNR computed from the resulting output was -7.70 dB. Then, a set of tests were run, exactly as in the previous set, one test for each additional CELP bit being protected and using the same mask file as in the previous tests. The changes in SEGSNRs for each CELP bit being protected relative to the SEGSNR with only the first 4 bits protected are shown in Figure 5.

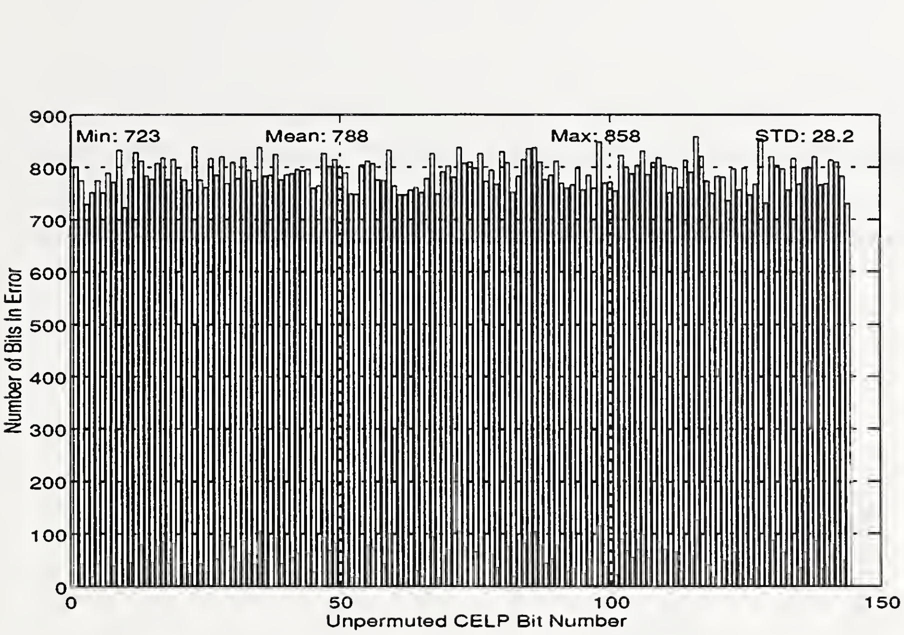

Figure 4: Number of Errors per CELP Bit

Using the same procedure shown above, 8 more sets of protection tests were conducted, each set starting with a group of protected bits determined by the results of the previous set, and then comparing the SEGSNRs when protecting one more CELP bit to SEGSNR of the first test in the set. Figures 6 to 13 show the results of these sets of tests.

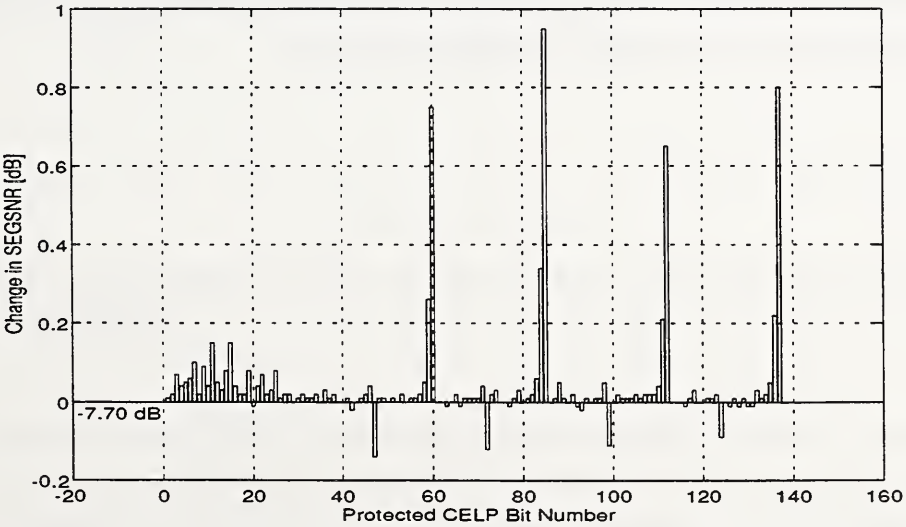

Figure 5: SEGSNR Improvement with Bits $\{61,86,113,138\}$,i^{th} Protected

Based on the results from Figure 5, the next set of tests protected bits 60-61, 85-86, 112-113 and 137-138, namely, the 2 most significant bits of the CBGAIN parameters. The results are shown in Figure 6, where the first test resulted in SEGSNR of -3.51 dB. It should be noted that the decrease in SEGSNR for bits 47, 72, 99 and 124 continued to appear in the last 2 tests, but with a reduced effect.

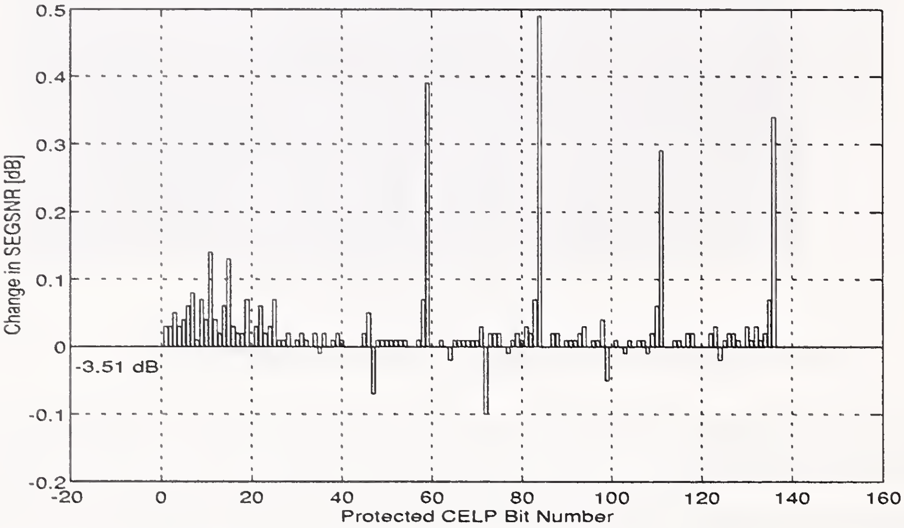

Figure 6: SEGSNR Improvement with Bits {61,86,113,138},{60,85,112,137},ith Protected

Based on the results from Figure 6, the next set of tests protected bits 59-61, 84-86, 111-113 and 136-138, namely, the 3 most significant bits of the CBGAIN parameters. The results are shown in Figure 7, where the first test resulted in SEGSNR of -1.52 dB. In this test, the next significant bits of the CBGAIN parameters show the highest increase in SEGSNR, some LSP bits show an increase as well, i.e., bits 11 and 15, and there is no decrease in SEGSNR for bit 124.

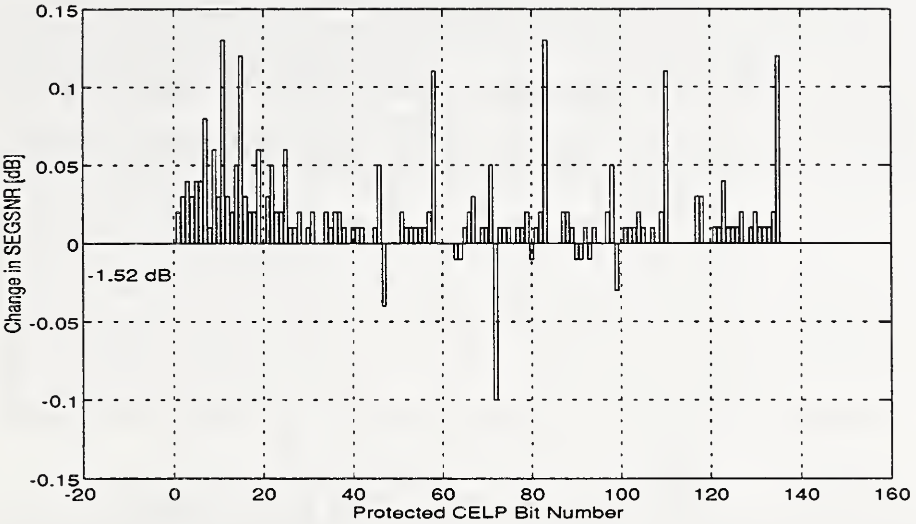

Figure 7: SEGSNR Improvement with Bits {61,86,113,138},{60,85,112,137},{59,84,111,136},ith Protected

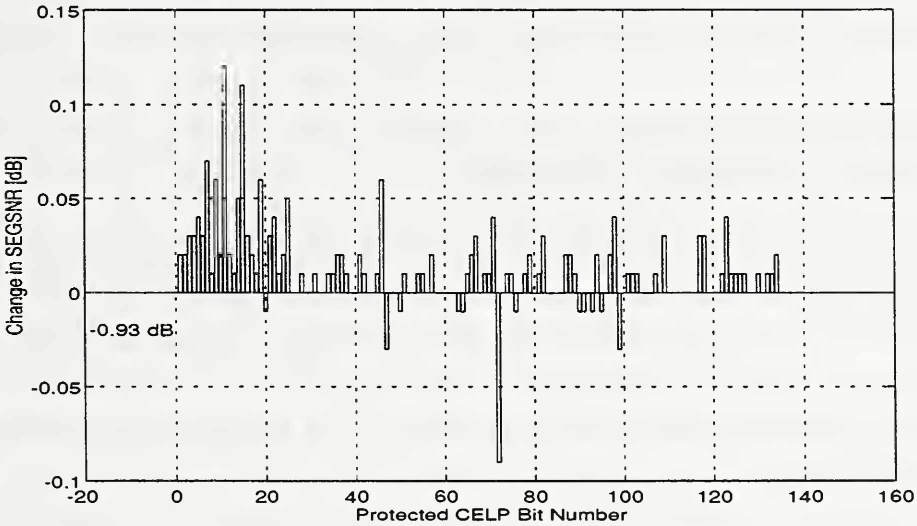

Figure 8: SEGSNR Improvement with Bits {61,86,113,138},{60,85,112,137},{59,84,111,136}, {58,83,110,135},ith Protected

Based on the results from Figure 7, the next set of tests protected bits 58-61, 83-86, 110-113 and 135-138, namely, the 4 most significant bits of the CBGAIN parameters. The results are shown in Figure 8, where the first test resulted in SEGSNR of -0.93 dB. In this test, some of the LSP most significant bits appear as most important, namely, bits 7, 11, 15, 19, 22 and 25. In addition, bits 46, 71, 98 and 123, the next to most significant bits in the pitch gain parameters, emerge as the next group of important bits.

Based on the results from Figure 8, the next set of tests protected bits 58-61, 83-86, 110-113 and 135-138, and bits 7, 11, 15, 19, 22 and 25. The results are shown in Figure 9, where the first test resulted in SEGSNR of -0.65 dB. In this test, some more of the LSP bits appeared to be important, namely, bits 10, 14 and 18. The pitch gain bits 46, 71, 98 and 123, showed about the same increase in SEGSNR. Note that, in this test, some more bits showed a decrease in SEGSNR when protected, but this was insignificant since for most of them it was within the resolution of the SEGSNR measurement (0.01 dB).

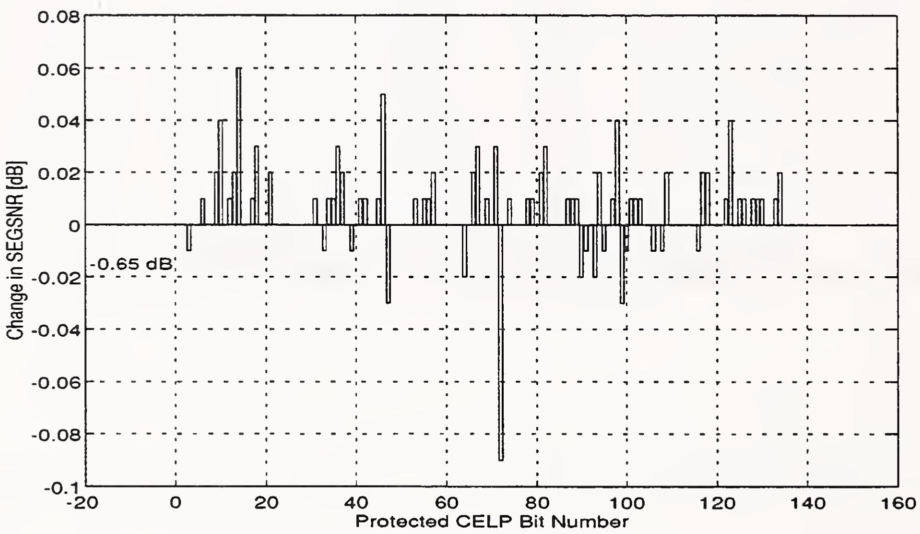

Figure 9: SEGSNR Improvement with Bits {61,86,113,138},{60,85,112,137},{59,84,111,136},{58,83,110,135},{7,11,15,19,22,25},ith Protected

In all the test sets so far, protecting the most significant bits of the pitch gain parameters caused a decrease in SEGSNR. At this point it became clear that bits 46, 71, 98 and 123 were to be protected. However, it seemed that the most significant bits of the pitch gain parameters had to be protected as well. First a test was run, with all the previous bits as well as bits 46, 71, 98 and 123 protected. Then, 4 more tests were run, protecting each one of bits 47, 72, 99 and 124. The resulting SEGSNRs are summarized in Table 1 below.

These results show that the selection of the next to most significant pitch gain bits was right, as the SEGSNR increased from -0.65 dB in the previous set of tests to -0.49 dB. However, when protecting the most significant pitch gain bits, SEGSNRs did not change significantly, except for bit 72, where it decreased by 0.02 dB. Hence, protecting the most significant pitch gain bits was delayed for later tests.

Table 1: Pitch Gain Bit Protection Test Results

Pitch Gain Bits Protected	**SEGSNR**
46, 71, 98 and 123 only	-0.49 dB
46, 71, 98, 123 and 47	-0.49 dB
46, 71, 98, 123 and 72	-0.51 dB
46, 71, 98, 123 and 99	-0.49 dB
46, 71, 98, 123 and 124	-0.46 dB

Based on the previous 2 sets of tests, the next set of tests protected bits 58-61, 83-86, 110-113, 135-138, 7, 10-11, 14-15, 18-19, 22, 25, 46, 71, 98 and 123. The results are shown in Figure 10, where the first test resulted in SEGSNR of -0.37 dB. In this test, for the first time, no bit caused a decrease in SEGSNR when protected (SEGSNR for bit 72 decreased by 0.01 dB, which, again, was the resolution of the SEGSNR measurement). At this point, the pitch delay and delta delay bits appeared to be important, namely, bits 40-42, 65-67, 92-94 and 117-119. Also, some of the pitch gain bits showed a significant increase in SEGSNR, namely, bits 45, 70, 97 and 122, but the most significant bits did not come out as important yet.

The next set of tests protected bits 58-61, 83-86, 110-113 and 135-138, and bits 7, 10-11, 14-15, 18-19, 22, 25, 46, 71, 98, 123, 40-42 and 92-94, namely, pitch delay bits were also protected. The results are shown in Figure 11, where the first test resulted in SEGSNR of -0.26 dB. In this test, the pitch gain bits showed an increase in SEGSNR, together with the least significant bits of the CBGAINs and some LSP bits.

Based on the results from Figure 11, the next set of tests protected bits 57-61, 82-86, 109-113, 134-138, and bits 7, 10-11, 14-15, 18-19, 22, 25, 40-42, 45-46, 70-71, 92-94, 97-98, 122-123. The results are shown in Figure 12, where the first test resulted in SEGSNR of -0.05 dB. In this test, the most significant bits of the pitch gain parameters, namely, bits 47, 72, 99 and 124, showed either an increase or no change in SEGSNR when protected. The delta delay bits and some more of the LSP bits showed an increase in SEGSNR as well. Finally, listening to the output audio file showed that the result was very intelligible and with almost no noise.

The next set of tests was based on the results from Figure 12. In this set the bits protected were bits 5-7, 9-11, 13-15, 18-19, 21-22, 25, 28, 31, 34, 40-42, 45-47, 57-61, 65-67, 70-72, 82-86, 92-94, 97-99, 109-113, 116-119, 122-124 and 134-138. The results are shown in Figure 13, where the first test resulted in SEGSNR of 0.237 dB.

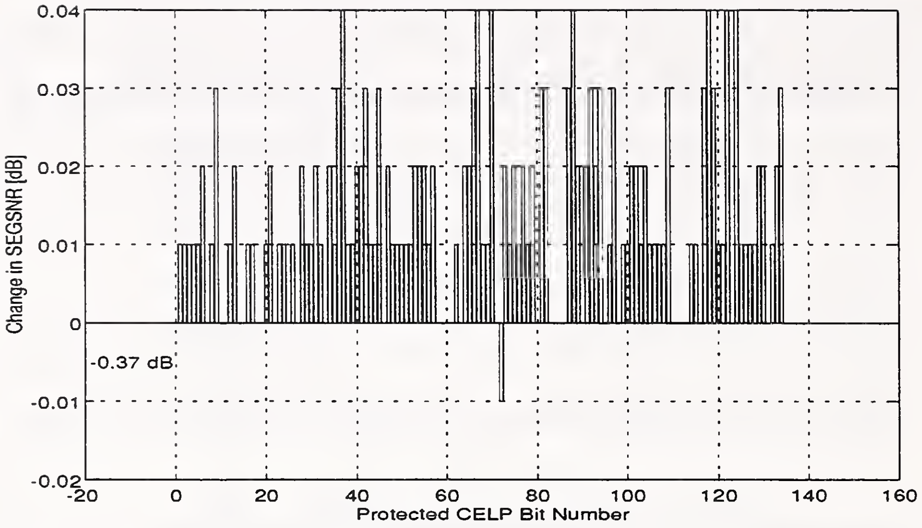

Figure 10: SEGSNR Improvement with Bits {61,86,113,138},{60,85,112,137},{59,84,111,136},{58,83,110,135},{7,11,15,19,22,25},{10,14,18,46,71,98,123},ith Protected

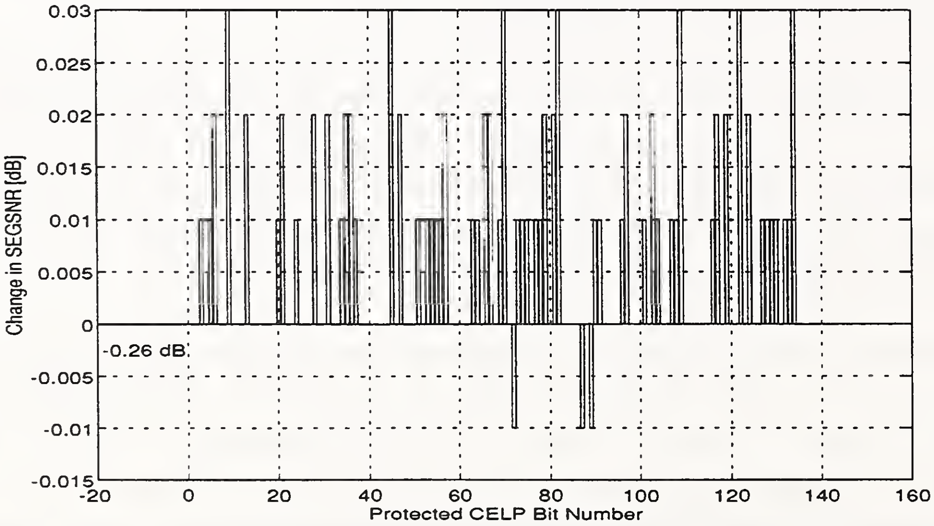

Figure 11: SEGSNR Improvement with Bits {61,86,113,138},{60,85,112,137},{59,84,111,136},{58,83,110,135},{7,11,15,19,22,25},{10,14,18,46,71,98,123},{40-42,92-94},ith Protected

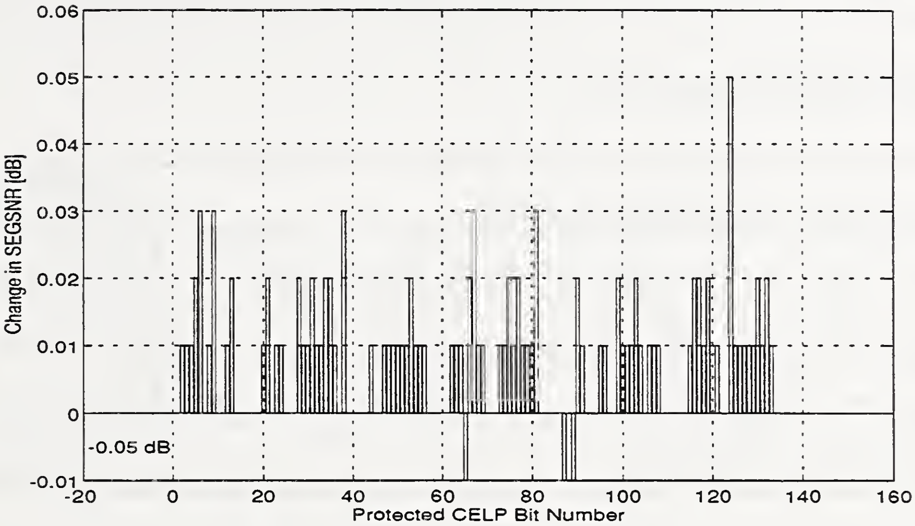

Figure 12: SEGSNR Improvement with Bits {61,86,113,138},{60,85,112,137},{59,84,111,136},{58,83,110,135},{7,11,15,19,22,25},{10,14,18,46,71,98,123},{40-42,92-94},{45-46,57,70,82,97,109,122,134},ith Protected

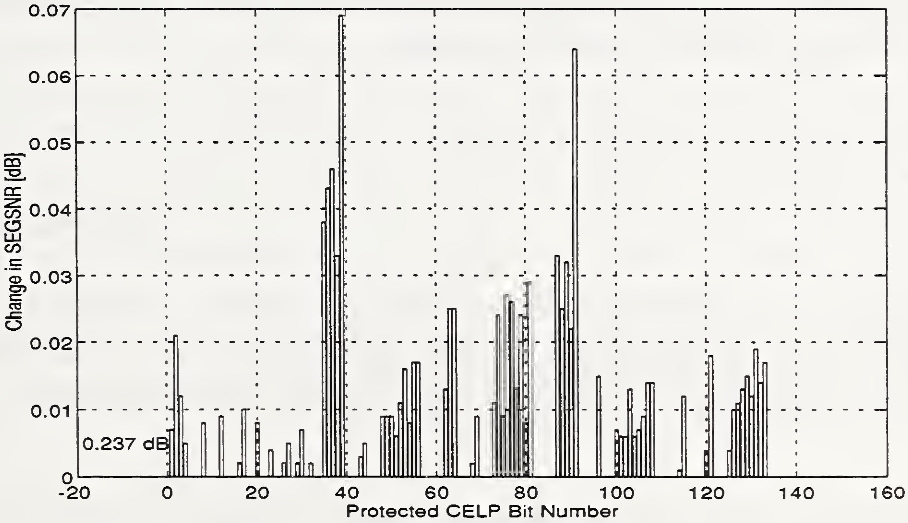

Figure 13: SEGSNR Improvement with the Bits of Figure 12 and {5-6,9,13,21,28,31,34,47,65-67,72,99,116-119,124},ith Protected

In order to protect a total of 84 bits, as was done in [8], 4 more tests were run, each one protecting another group of bits, but single bit sensitivity was not checked. In addition, the output audio files were saved for the listening tests. The results are shown in Table 2.

Table 2: Final Bit Protection Test Results

Additional CELP Bits Protected	**SEGSNR**
35, 36, 37, 38, 39, 87, 88, 89, 90, 91 (pitch delay)	0.862 dB
62, 63, 64, 114, 115 (delta delay)	1.167 dB
96, 121 (pitch gain)	1.222 dB
2, 3, 17, 20, 30 (LSPs)	1.307 dB

4.4 Conclusions from Bit Protection Tests

Following the order of protecting the different bits in the tests described in this section, it can be seen that the CBGAIN bits are the most sensitive to errors, and should get the most powerful protection against channel errors. These results match the results from the bit error tests described in the previous section. Following in importance are some of the LSP bits, together with some of the pitch gain bits. The following bits in importance are the pitch delay and delta delay bits. The codebook index bits show the least sensitivity to errors, as found in the bit error tests.

Comparing the results of the bit error tests and bit protection tests, the conclusion is that the CELP bits to protect should include the CBGAIN, pitch gain, pitch delay and delta delay parameters, but should not include the codebook index parameters. The relative importance of the bits should be defined by the bit protection tests since these were much more rigorous, and used simulated channel error bit masks. The various bits in each group of parameters have different sensitivities, so different protection levels are necessary. For example, in the pitch gain bits, the most significant bits (47, 72, 99, 124) are more sensitive to errors, but the least significant bits (43, 68, 95 and 120) are not sensitive and can even be left without protection.

The results from the tests described above were used to design two error correction codes, which are described in the following two sections.

5. Design of Error Correction Code

In this section we will describe how the error correcting codes used for protecting the CELP frame bits have been selected. This will pave the way for a meaningful comparison between different systems in the next section.

5.1 Basic Assumptions

Recall that each 30 msec CELP frame contains 144 bits, leading to a bit rate of 4800 bps. Clearly, the more protection added by error correcting codes the more robust the system will become in the presence of channel noise but at the same time the overall bit rate will be higher. In this work, we have limited ourselves to one specific case where the overall bit rate (after adding the protection bits) is 7200 bps. This is done for two reasons: (i) The encoding rate of 7200 bps is compatible with the U.S. Government Land Mobile Radio system in which the proposed bit rate is 9600 bps with 2400 bps set aside for framing, leaving 7200 bps for speech coding and error protection; (ii) the speech encoder proposed by the NSA, and against which we will make comparisons, is also designed for operation at 7200 bps.

The NSA encoder, as described in [8], uses RS codes to protect 84 bits in each frame of 144 CELP output bits, leaving 60 bits unprotected. The 84 bits are protected by an additional 72 error protection bits. This will map every CELP frame of 144 bits to a frame of 144+72=216 bits, resulting in a rate of 7200 bps.

Following the same spirit as in [8], we will use RS codes to protect 84 bits in each CELP frame, leaving 60 bits unprotected. The main difference between the systems proposed here and that of [8] is the use of unequal error protection in the proposed systems. The details are provided below.

5.2 The NSA Encoder [8]

The encoder proposed in [8] uses RS codes over GF(16). Specifically, the code used was a (13,7) code over GF(16) thus protecting 28 bits by adding 24 error protection bits. Three such codes, denoted (13,7)4, were used to encode 84 CELP bits. To be precise, the three codewords were used as follows:

Codeword Number	CELP Parameters
1	Odd LSPs(11 bits), PD3(8 bits), DD4(6 bits), PG4(3 bits)
2	Even LSPs(11 bits), PD1(8 bits), DD2(6 bits), PG2(3 bits)
3	CBG1, CBG2, CBG3, CBG4(each 5 bits), PG1(4 bits), PG3(4 bits)

The acronyms used above are taken from [8]. CBG stands for codebook gain, PD for pitch delay, PG for pitch gain, DD for delta delay and LSP for line spectrum pairs. We must emphasize that in the above scheme the same level of protection is used for all bits which are protected.

In the simulation of the NSA system, the bits were allocated to the three codewords exactly in the order shown above, assigning 4 bits at a time to the symbols in the codewords.

5.3 The NIST Proposed Encoders

Based on the bit sensitivity analyses of Section 4, it is clear that the bits representing CELP encoded speech exhibit different levels of sensitivity (SEGSNR) to transmission noise. Therefore, it is natural to think of an error protection scheme which applies an unequal level of protection to the CELP encoded speech such that more protection is applied to those bits that exhibit a higher sensitivity to channel noise and less to those bits that are not as sensitive. If this is done correctly, the extra bits used for protection will be used where they count most.

Of course, there are many combinations that could be tried for unequal error protection. In an effort to strike a balance between added system complexity and more efficient utilization of error correction coding, we have considered two schemes. In both schemes, RS codes over GF(16) are used to be in line with the scheme in [8].

In general, for an (n,k)4 RS code, the smaller the rate, r=k/n, the more powerful the code. We have used the results of Section 4 on the sensitivity of CELP encoded speech to channel noise to propose two specific schemes for unequal error protection. These two schemes use RS codes. The definition of RS codewords are provided below in terms of CELP bits . Note that the bit numbers represent bit numbers in the CELP frame as defined in [2].

Scheme 1:

Encode using 3 codewords with (11,5)4, (13,7)4 and (15,9)4 RS codes, as shown below:

Symbol No.	Parameter	Bit No. 1	Bit No. 2	Bit No. 3	Bit No. 4
1	CBG	61	86	113	138
2	CBG	60	85	112	137
3	CBG	59	84	111	136
4	CBG	58	83	110	135
5	LSPs	3	11	19	25

Symbol No.	Parameter	Bit No. 1	Bit No. 2	Bit No. 3	Bit No. 4
1	LSPs	7	15	22	28
2	PG	47	72	99	124
3	PG	46	71	98	123
4	LSPs	2	10	18	24
5	LSPs	6	14	21	27
6	PD	41	42	93	94
7	PG	45	70	97	122

Symbol No.	Parameter	Bit No. 1	Bit No. 2	Bit No. 3	Bit No. 4
1	P/D D	40	67	92	119
2	CBG	57	82	109	134
3	LSPs	9	17	23	31
4	LSPs	5	13	20	34
5	DD	65	66	117	118
6	PD	38	39	90	91
7	PD	36	37	88	89
8	DD	63	64	115	116
9	P/D D	35	62	87	114

Scheme 2:

Encode using 4 codewords with (9,3)4, (11,5)4, (10,6)4 and (9,7)4 RS codes, as shown below:

Symbol No.	Parameter	Bit No. 1	Bit No. 2	Bit No. 3	Bit No. 4
1	CBG	61	86	113	138
2	CBG	60	85	112	137
3	CBG	59	84	111	136

Symbol No.	Parameter	Bit No. 1	Bit No. 2	Bit No. 3	Bit No. 4
1	CBG	58	83	110	135
2	LSPs	3	11	19	25
3	LSPs	7	15	22	28
4	PG	47	72	99	124
5	PG	46	71	98	123

Symbol No.	Parameter	Bit No. 1	Bit No. 2	Bit No. 3	Bit No. 4
1	LSPs	2	10	18	24
2	LSPs	6	14	21	27
3	PD	41	42	93	94
4	PG	45	70	97	122
5	P/D D	40	67	92	119
6	CBG	57	82	109	134

Symbol No.	Parameter	Bit No. 1	Bit No. 2	Bit No. 3	Bit No. 4
1	LSPs	9	17	23	31
2	LSPs	5	13	20	34
3	DD	65	66	117	118
4	PD	38	39	90	91
5	PD	36	37	88	89
6	DD	63	64	115	116
7	P/D D	35	62	87	114

6. Comparison of the NSA System to NIST Systems

6.1 Purpose of Tests

These tests were intended to compare the different error correction codes, in terms of their error correction capabilities, using the same number of protection bits for all codes, but allocating the CELP bits to different codewords.

The NSA system of included 3 codewords of (13,7)4 over GF(16). The first NIST system included 3 codewords of (11,5)4, (13,7)4 and (15,9)4 over GF(16), and is referred to as NIST System-1. A slight variation of this system was also tested, in which all 4 CBGAIN bits in the first codeword were grouped into the same symbols. This system is referred to as NIST System-1A. Finally, the second NIST system included 4 codewords of (9,3)4, (11,5)4, (10,6)4 and (9,7)4 over GF(16), and is referred to as NIST System-2. Since NIST System-1 has produced SEGSNR results which are better than those of NIST System-2, emphasis is placed on the first system.

6.2 Method of Testing

For each system, all the stages in the communication system were simulated in software, as illustrated in Section 1. The CELP simulation was exactly the same as in the previous tests. Since the bits to be protected in each CELP frame were allocated to different codewords in all the systems, it was necessary to split each CELP frame and group the bits into the appropriate codewords as well as an unprotected group of bits. Each group of bits was then encoded by the RS-encoder software, which produced the channel encoded data. The unprotected group of bits were left unchanged.

The masking of the channel data used bit error masks of 7200 bits per second. This process involved collecting all RS-encoded bits and the unprotected bits for each CELP frame, and then masking the bits with the corresponding bit error mask. In order to save CPU cycles, simulation of symbol interleaving was actually implemented by permuting the mask bits before the masking operation, instead of permuting the data bits, masking and then unpermuting the data bits to their original form, which was a longer process.

Symbol interleaving was used in both systems in order to overcome burst errors characteristic of wireless channels. This was done by spreading the symbols from each codeword as evenly as possible within the available symbols used for each RS-encoded CELP frame. This maximized the space between adjacent symbols in a codeword, so that the probability of an error in 2 symbols of the same codeword was reduced. Note that such an interleaving can be done over more than one frame, but then a correspondingly longer delay is introduced in the processing. Since all systems used 4 bit symbols, interleaving was also based on groups of 4 bits in all systems.

For the NSA system there were 3 codewords of 13 symbols and 15 symbols for the unprotected bits. The interleaving allocation was relatively simple, since all 4 symbol groups had almost the same number of symbols.

Figure 14 shows an example of the interleaving scheme used for NIST System-1. The symbols numbered "1"-"3" belong to the corresponding RS codewords groups and the symbols which are numbered "4" belong to the group of unprotected bits, all from one CELP frame. Time advances from left to right and from top to bottom. Since the 3 codewords included 11, 13 and 15 symbols and the unprotected bits included 15 symbols, the interleaving allocation started by allocating the 11 symbols from the first codeword as evenly as possible within the available 54 symbols. This resulted in 10 symbols separated by 4 other symbols, and 1 symbol separated by 3 other symbols. Then the 13 symbols from the second codeword were allocated into the remaining 43 symbols using the same method. This procedure was repeated for the 15 symbols from the third codeword and the 15 unprotected symbols.

1	2	3	4	3	1	2	4	3	4	1	2	3	4	3	1
2	4	3	4	1	2	3	4	2	1	3	4	2	3	1	4
2	3	4	1	2	3	4	2	1	3	4	2	3	1	4	2
3	4	1	2	3	4										

Figure 14: Symbol Interleaving for NIST System-1

Following the masking process, the RS-decoder software was used to decode each group of masked bits. Then the decoded bits and the unprotected bits were reassembled into a CELP frame, which was then synthesized in the same manner as in the previous tests.

Measurement of quality was based on SEGSNR, again with the input audio as the signal. In addition, actual listening to the resulting output speech was performed, comparing the NSA system to the best NIST system, namely, NIST System-1A.

In order to test different channel conditions, 4 bit error masks were used, as shown in Table 3. These were generated by the NSA radio channel simulator.

Table 3: List of Channel Bit Error Masks

Receiver Speed	Channel SNR	Bit Error Rate
5 MPH	18 dB	3%
30 MPH	14 dB	5%
60 MPH	15 dB	5%
3 MPH	10 dB	10%

6.3 Summary of Results

Table 4 shows the SEGSNRs for each one of the bit error masks used in 4 tests. In all tests, the smoothing was performed in the CELP synthesis.

Table 4: SEGSNRs of Error Correction Systems

Test Conditions (BER/Speed)	NSA System	NIST System-1	NIST System-1A	NIST System-2
3% BER (5 MPH)	-0.978 dB	-0.475 dB	-0.261 dB	-0.270 dB
5% BER (30 MPH)	-2.591 dB	-2.263 dB	-1.829 dB	-2.093 dB
5% BER (60 MPH)	-1.965 dB	-1.507 dB	-1.244 dB	-1.437 dB
10% BER (3 MPH)	-6.883 dB	-6.457 dB	-6.251 dB	-5.970 dB

As is evident from Table 4, all NIST systems resulted in SEGSNRs slightly higher than those for NSA System, and under all test conditions. In addition, SEGSNRs for NIST System-1A were the highest under all test conditions. Therefore, it can be concluded that encoding CELP bits with unequal error protection increases the overall quality, and that grouping the CBGAIN bits into the same symbols increases the signal quality even further. Comparing SEGSNRs for the two 5% BER tests, all tests show that SEGSNR at a higher speed is higher, when the BER is fixed. This can probably be explained by the fact that at higher speeds the bursts of noise caused by multipath fading are more frequent but they are shorter in length. Therefore, the number of symbols in error per codeword is probably reduced, allowing the RS code to correct more errors overall, thus producing a better signal quality.

Following SEGSNR measurements, listening tests were also used in order to get subjective measurements of the best NIST System (NIST System-1A) and the NSA System. These tests were conducted by playing pairs of segments from the output audio files resulting from the same test conditions but using different systems. The listeners did not know in advance the order in which the audio files were going to be played, nor did they know what error mask was going to be played, thus making the tests as fair as possible. Scoring was done by having each listener mark which audio files sounded better in terms of noise or blasts, general intelligibility and number of words missed. Then the results from all listeners were compiled.

The results from the listening tests showed that at the low bit error rate of 3% both systems sounded very good with about the same quality. At the high bit error rate of 10% both systems sounded very bad and it was hard to decide which one was better. At both 5% bit error rates NIST System-1A sounded slightly better. In addition, when comparing the two different error masks for the 5% BER, it was found again that the one at higher receiver speed resulted in better quality. These results match the objective measurements using SEGSNR.

7. Conclusions

The main conclusion from this work is that the unequal protection scheme selected in this work, with more protection given to more sensitive CELP bits, such as the codebook gain bits, is slightly better than an equal protection scheme, using both objective and subjective quality measures. Further research is needed in order to achieve better performance improvements, by using different error correction codes, such as convolutional codes. It is also concluded that the bit protection sensitivity tests were effective in determining the protection levels of bits.

8. References

[1] White Paper on a National Wireless Performance Benchmarking Program, NIST/NTIA, Wireless Benchmarking Workshop, Gaithersburg, MD, August 1994.

[2] Federal Standard 1016, "Telecommunications: Analog to Digital Conversion of Radio Voice by 4800 bit/second Code Excited Linear Prediction (CELP)," National Communications System, Office of Technology and Standards, Washington DC, 20305-2010, 14 February 1991.

[3] Bracha, E., "Real-Time Simulation of Wireless Personal Communication Systems," Distributed Systems Group Internal Report, Advanced Systems Division, NIST, Gaithersburg, MD, November 1993.

[4] Bracha, E., "Software Simulation of Outdoor Mobile Radio Channel Effects on Direct Audio Samples," Distributed Systems Group Internal Report, Advanced Systems Division, NIST, Gaithersburg, MD, December 1993.

[5] Bracha, E., "Software Simulation of Outdoor Mobile Radio Channel Effects on CELP Encoded Audio Samples," Distributed Systems Group Internal Report, Advanced Systems Division, NIST, Gaithersburg, MD, January 1994.

[6] Rahikka, D., Tremain, T., Welch, V., Campbell, J. Jr., "CELP Coding for Land Mobile Radio Applications," ICASSP '90, Albuquerque, New Mexico, April 1990.

[7] Shi, H., Ho, P., Cuperman, V., "Combined Speech and Channel Coding for Mobile Radio Applications," ICWC '92, February 1992.

[8] Ray, D., Rahikka, D., "Reed-Solomon Coding for CELP EDAC In Land Mobile Radio," ICASSP '94, Adelaide, Australia, April 1994.

[9] Garofolo, J.S., Lamel, L.F., Fisher, W.M., Fiscus, J.G., Pallett, D.S., Dahlgren, N.L., "DARPA TIMIT Acoustic-Phonetic Continuous Corpus CD-ROM," NISTIR 4930, February 1993.

[10] AT&T and Hughes Network Systems, Technology Assessment Radio Channel Simulator, April 1993, Personal communication.

[11] NSA, VHFBER Radio Channel Simulator, Personal Communication.

Acknowledgements

The authors would like to express their gratitude to Wayne McCoy and Shukri Wakid from NIST for providing the opportunity to work at the Digital Signal Processing Group of the Advanced Systems Division, where the research reported in this paper has been performed. Thanks are also due to Robert Toense for his help throughout, and to Mary Ruhl for her help in producing this report.

Finally, the authors would like to thank AT&T and Hughes Network Systems for providing the software of their radio channel simulator (identified as the AT&T/Hughes Simulator in this paper), as well as to the NSA for providing the software of their VHF radio channel simulator and the software for error correction coding, without which it would have been virtually impossible to complete this work in the given time-frame.