Robust Tree Coding Combined with Harmonic Scaling of Speech at 4.8 Kbps

Sang Won Kang*, In Sung Lee*, Kyong Ho Han* *Regular Members*

견실한 배음 축척과 결합된 4.8 KBPS 트리 음성부호기

正會員 姜 尙 遠* 正會員 李 寅 誠* 正會員 韓 敬 浩*

ABSTRACT

Efficient speech coders using tree coding combined with harmonic scaling are designed at the rate of 4.8 kilobits/sec (kbps). A time domain harmonic scaling algorithm (TDHS) is used to compress input speech by a factor of two. This process allows the tree coder to have 1.5 bits/sample for 4.8 kbps in the case of a 6.4 kHz sampling rate. In the backward adaptive tree coder, there are three components of the code generator, including a hybrid adaptive quantizer, a short-term predictor and a pitch predictor. The robustness of the tree coder is achieved by carefully choosing the input of the short-term predictor adaptation. Also, inclusion of a smoother in the pitch predictor improves the error performance of tree coder in the noisy channel. Subjectively, tree coding combined with TDHS provides good quality speech at 4.8 kbps.

要 約

본 논문에서는 음성 신호의 4.8 Kbps에서 효율적인 배음 축척과 결합된 트리 부호기를 실현한다. 음성 신호를 2대 1 압축하기 위해 TDHS 알고리즘을 사용한다. 이 과정은 4.8 Kbps에서 6.4 KHz 샘플링율을 적용하면 트리 부호기에 1.5 비트/샘플을 할당할 수 있다. 트리 부호기의 견실성은 short-term 예측기의 적응시 사용되는 입력 신호를 효율적 선택함으로써 개선되어진다. 또한 채널에서 전송에러시 트리 부호기의 성능은 피치 예측기에 스무더를 부가함으로써 개선된다. 배음 축척과 결합된 트리 부호기는 4.8 Kbps 전송률에서 좋은 질의 음성을 출력한다.

I. Introduction

The speech communication is the transfer of information from one person to another via speech. In a digital communication system the speech signals is coded into a bit stream representation, transmitted over a channel, and converted back into an audible signal. The transmission of speech data at low bit rates is becoming increasingly

*韓國電子通信硏究所
ETRI
論文番號 : 93-182

needed for many applications such as narrowband digital transmission, storage of speech data and future demands for personal communication networks.

The frequency scaling is a useful method for reducing the bandwidth of the speech signal in order to assign more bits to a waveform coder. Malah developed a time domain harmonic scaling (TDHS) algorithm [1], which is a simple and efficient method to reduce the coding bit rate by half or one-third of the original ones. The TDHS algorithm gave a significant reduction of bit rate with little loss in speech quality when it was incorporated whith some waveform coders. TDHS was combined with continuously variable slope delta modulation (CVSD) [2] and subband coding [3] to achieve acceptable communication quality at 7.2 kbps and 9.6 kbps, respectively.

One promising approach for improving the performance at lower rates is to use multipath tree search strategies in contrast to conventional single-path coders including differential pulse code modulation, delta modulation, and adaptive predictive coding. In multi-path tree coding, future input samples are considered as well before a decision is made about the optimum present codeword to be released. Tree coding has been applied to speech signal by several researchers because of its possible performance superiority to single-path coders. Anderson and Bodie applied the (M, L) algorithm to speech signal in order to reduce the computational burden of tree search [4]. Gibson and Chang developed a multitree coder [5] which consists of different number of branches interleaved with each other in the code tree. This structure preserved high frequency components of the original well and allowed flexibility to construct fractional rate tree coders.

In this paper, a speech coder that combines tree coding with TDHS is designed at the rates of 4.8 kbps. The structure of code tree has a multitree or vector tree for the fractional rate coder. The code generator is designed with the consideration of the input characteristics of frequency compressed speech. For real applications of speech coding, it is important to design for the robustness to the transmission errors. Therefore, the tree coder is designed to achieve the robustness to channel errors as well as to maintain the high quality of speech in the error free channel.

II. Time Domain Harmonic Scaling

Frequency scaling is a useful method for reducing the bandwidth of speech. An efficient and simple algorithm for frequency scaling of the speech signal is provided by time domain harmonic scaling (TDHS) [1]. A short-time Fourier transform of two or three periods of voiced speech shows a strong harmonic structure. The bandwidth of each harmonic is usually narrow compared to the spacing between harmonics. Therefore, even if the spectrum around each harmonic is shifted down in frequency by a factor of two or three, little information is lost because aliasing occurs only in the spectral valleys. After isolating the different pitch harmonics, each harmonic is modulated to its designated new location by using filter bank analysis.

In this paper, the compression factor is chosen to 2. The input speech is compressed by a factor of two in the transmitter and expanded in the receiver by using TDHS. At the transmitter, the basic concept of compression is to compress two pitch period speech samples into a single pitch period with the same time duration, but at half the sampling rate. At the receiver, two pitch period samples are extracted by interpolation of

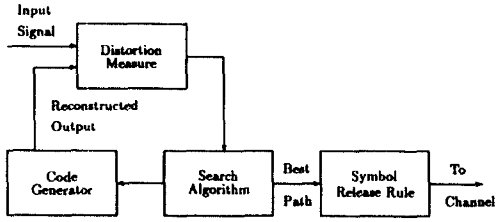


Fig. 1. Elements of a Tree Coder

neighboring blocks of the present pitch period block. The selection of the window function affects the performance of TDHS. The Hanning window among various window functions has the best performance [6]. A triangular window can be chosen because of its simplicity. It requires one multiplication and two additions to have one compressed sample.

Ⅲ. TREE CODING OF SPEECH

A block diagram of a typical tree coder is shown in Fig. 1. It consists of four functional elements: a code generator, a distortion measure, a search algorithm, and a symbol release rule. The code generator produces reconstructed output sequences according to all possible path maps. For each path map, the excitation sequence is generated by the output of a scalar quantizer with certain statistics, or by stochastic codebook, or by a codebook of a vector quantizer designed for the innovation sequence. The generated excitation sequence is multiplied by a gain term and then filtered out through the synthesizer to produce candidate output signals. This reconstructed output sequence is transmitted to the distortion measure block to calculate the distortion between the source sequence and the reconstructed candidate output sequence to the depth L. The use of the frequency weighted distortion measure helps it to reduce the perception of quantization noise. In order to reduce the heavy computational load of exhaustive search, the (M, L) algorithm is used [4]. A symbol release rule specifies how many symbols are released to the channel after deciding the minimum distortion path. In our work, (M, L) algorithm with $M=8$, $L=16$ and the incremental fixed symbol realease rule are used.

A. Gain Adaptation

The gain term should be adapted to the input signal because the speech signal is nonstationary and has relatively large input dynamic range. For the multi-tree coder, the hybrid adaptive quantizer and delta modulation algorithm are used in tree coding. The hybrid gain adaptation rule combining the instantaneous adaptation and syllabic adaptation is given by

$$\Delta(n+1)=M(n)\,\delta^{\gamma}(n)\,\Delta^{\beta}(n) \tag{1}$$

with $\delta(n)=\alpha\delta(n-1)+(1-\alpha)\,\theta|u(n)|$ where $\alpha=0.9$ controls the effective memory of the estimator, θ is a constant, and $u(n)$ is the latest quantizer output. In the hybrid quantizer, $\beta=50/64$, $\gamma=13/64$, and $\theta=1.254$ are chosen.

For the vector tree coder, the gain term is calculated using the modified algorithm employed in [7]. Let x be the root mean square (RMS) value of a code vector. The gain adaptation algorithm is

$$\Delta(n+1)=f(x)\,\Delta^{\beta}(n) \tag{2}$$

with multiplier function $f(x)$ given by

$$f(x)=\begin{cases}\delta^{(\gamma)}(n)\,e^{\frac{5}{3}c_2(x-\frac{3}{5})} & \text{if } 0\le x<x_1\\ \delta^{(\gamma)}(n)\,e^{\frac{5}{6}c_1(x-\frac{3}{5})} & \text{if } x_1\le x<x_2\\ \delta^{(\gamma)}(n)\,M_{max} & \text{if } x\le x_2\end{cases}$$

with $\delta(n)=\alpha\delta(n-1)+(1-\alpha)\,\theta|u(n)|$ where $\alpha=0.9$, $\theta=1.3$ and $u(n)$ is an element of the codebook vector. The long-term gain adaptation is included in the multiplier function. Suitable parameter values are $\beta=0.853$, $\gamma=0.13$, $c_1=ln(M_{max})$, $c_2=-ln(M_{min})$, $M_{max}=1.8$, $M_{min}=0.8$, $x_1=0.6$, and $x_2=1.8$. The gain adaptation with every sample update of $\delta(n)$ provides a slight performance increase over that with a fixed value δ.

B. Short-term Predictor

In speech signals, there is a correlation between adjecent samples introduced by the speaker vocal tract shape. The short-term all-pole predictor is defined by

$$A(z)=\sum_{k=1}^{N} a_k z^{-k} \tag{3}$$

where a_k is the short-term predictor coefficient and N is the order of the predictor. The coefficients of the all-pole predictor can be backwardly adapted by using numerous adaptive filter algorithms. It was reported that the LSL algorithm has the best performance in objective and subjective tests [8]. Therefore, the recursive LSL is chosen for the backward adaptation for short-term predictor.

It is well-known that an accurate model of the vocal tract should contain zeros as well as poles [9]. Moreover, the inclusion of zeros in ADPCM allows the predictor to be more responsive to changes of input signal and more robust to channel errors than an all-pole structure. The all-zero predictor has transfer function $B(z)=\sum_{j=1}^{M} b_j z^{-j}$, so that the predicted value of pole-zero predictor is given by

$$\hat{y}(n|n-1)=\sum_{i=1}^{N} a_i \hat{y}(n-i)+\sum_{j=1}^{M} b_j r_o(n-j) \tag{4}$$

The transfer function of pole-zero predictor in a code generator is

$$\frac{\hat{Y}(z)}{R_o(z)}=\frac{1+B(z)}{1-A(z)} \tag{5}$$

The all-zero adaptation algorithm of CCITT is used for the all-zero predictor adaptation [10]. The all-zero adaptiation of CCITT is

$$b_j(n+1)=\frac{127}{128} b_j(n)+\frac{1}{128} sgn[e_q(n-j)]sgn[e_q(n)],$$
$$j=1, \cdots, M \tag{6}$$

C. Imput to the Short-term Predictor Adaptation

It is important to choose the input to the adaptation in the pole-zero predictor for the robustness of tree coding to channel errors. The conventional input of the all-pole predictor adaptation is the reconstructed output signal. The output driven adaptation produces the best performance in the error free channel, but it is very sensitive to channel errors because the output signal has a long memory. The output signal is synthesized by the all-pole predictor which has the infinite impulse response. One solution to the mistracking problem of a short-term predictor due to the long memory of the output signal is to use the quantized residual signal instead of the output signal. The residual driven adaptation becomes very robust even in the presence of a high transmission error rate. However, the performance degradation of the residual driven adaptation in an ideal channel is inevitable.

Recently, one good tradeoff between the output signal driven adaptation and the residual signal driven adaptation was proposed [7]. The residual signal is shaped by an all-zero filter to resemble the reconstructed output signal. The all-zero shaping filter is obtained by truncating the infinite impulse response of an all-pole short-term synthesizer or by estimating the moving average model through the finite autoregressive model [11]. If the all-zero shaping filter is given by $1+\sum_{k=1}^{P} c_k z^{-k}$, then $C(z)$ is chosen to satisfy

$$\frac{1}{1-A(z)} \cong 1+C(z) \tag{7}$$

In our pole-zero structure for a short-term predictor, the all-zero shaping filter is obtained by truncating the impulse response of the pole-zero transfer function. The all-zero shaping filter is chosen to satisfy,

$$\frac{1+B(z)}{1-A(z)} \cong 1+D(z) \tag{8}$$

Then, the shaping filter coefficients, d_k are obtained by

$$d_k=\begin{cases} \sum_{j=1}^{N} a_j d_{k-j}+b_k & 1\le k\le M \\ \sum_{j=1}^{N} a_j d_{k-j} & M<k\le P \end{cases} \tag{9}$$

where M is the order of the all-zero predictor. The filtered residual signal is given by

$$\tilde{e}_d(n)=u(n)+\sum_{k=1}^{P} d_k u(n-k) \tag{10}$$

Fig. 2 shows the configuration for the filtered residual signal as the input of the all-pole short-term predictor adaptation. The shaping filter is given by the approximation of transfer function $\frac{1+B(z)}{1-A(z)}$.

The use of the filtered residual signal shaped by $1+D(z)$ provides 1.5 dB gain in SNRSEG over the use of the residual alone and exhibits the equivalent performance to an output driven adaptation in the error free channel. The SNRSEG of each method is graphically compared in Fig. 3. The filtered residual driven adaptation shows substantial improvement over the output driven adpatation in noisy channel performance. The increase of SNRSEG in the noisy channel is 3 dB at BER of 10^{-3} and 4 dB at BER of 10^{-2}. At BER of 10^{-4}, the residual driven adaptation achieves the best performance.

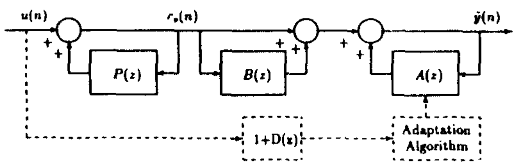


Fig. 2. Filtered Residual Driven Method for All-Pole Predictor in Pole-Zero Short-term Predictor

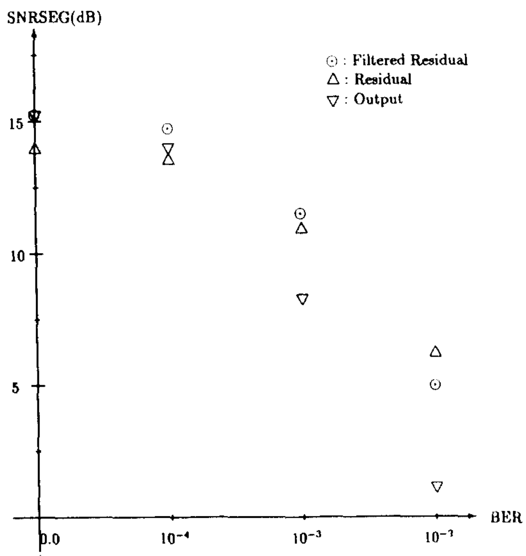


Fig. 3. Performance Comparison of All-Pole Adaptation Inputs

D. Pitch Predictor

In addition to short-term redundancies, there are long-term redundancies between samples with pitch period separation in a speech signal. The three tap long-term (pitch) predictor has the transfer function,

$$P(z) = \beta_{-1} z^{-(M_1-1)} + \beta_0 z^{-M_1} + \beta_1 z^{-(M_1+1)} \qquad (11)$$

where M_1 is the pitch period and β_{-1}, β_0, β_1 are long term predictor coefficients. In a backward block adaptation for the pitch predictor, the pitch period and pitch predictor coefficients are estimated from a block of previously reconstructed pitch synthesizer outputs. The pitch period estimate M_1 is determined by searching the lag with which the normalized correlation function is maximized. The pitch period M_1 is found in some bounded range. After the pitch period M_1 is decided, pitch predictor coefficients are calculated by solving the Wiener-Hopf equation.

A method which the pitch predictor parameters are recursively updated at every sample between the block adaptation was developed by Pattigrew and Cuperman [12]. This recursive backward adaptation consists of pitch period tracking and pitch predictor coefficient adaptation. The pitch predictor coefficients are updated by the gradient algorithm as

$$\beta_k(n) = \lambda\beta_k(n-1) + \frac{\mu_s}{\hat{\sigma}_{e_q}(n)\,\hat{\sigma}_{r_o}(n)}\, e_q(n)\, r_o(n-M_1+k)$$

$$k = -1,\ 0,\ 1 \qquad (12)$$

where the leakage factor λ is given by 0.95 and the constant step size μ_s is given by 0.06. The variances of $\hat{e}_q(n)$ and $\hat{r}_o(n)$ are updated by

$$\hat{\sigma}_x^2(n) = \lambda\hat{\sigma}_x^2(n-1) + (1-\lambda)\, x^2(n) \qquad (13)$$

Since the all-pole structure of the pitch synthesizer is very sensitive to channel errors, the input of the pitch predictor is given by the interpolation of neighborhood samples. The smoother with 3 taps has a transfer function $S(z)=s_1z^{-1}+s_0z^0+s_{-1}z^1$. The adaptation scheme of the pitch predictor is illustrated in Fig. 4. The pitch predictor becomes more robust if the coefficients of the smoother are chosen to implement the function of a low pass filtering. The performance comparisone of four pitch adaptation methods which include Cuperman's hybrid adaptation, Cuperman's hybrid adaptation without the pitch tracker, a hybrid adaptation with a fixed smoother, and a hybrid adaptation with a variable smoother are presented in Fig. 5

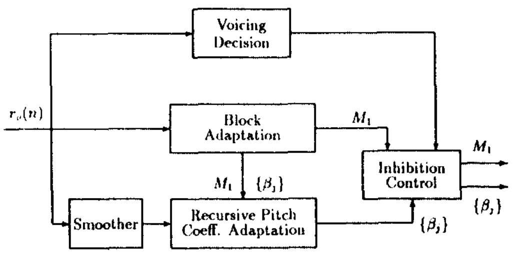


Fig. 4. Block Diagram for Pitch Predictor Adaptation with a Smoother

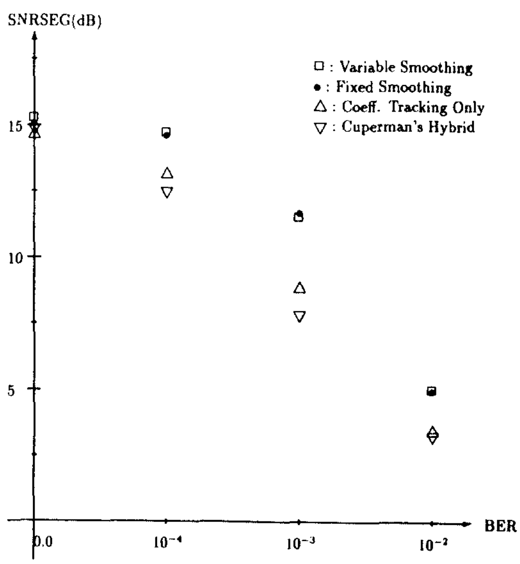


Fig. 5. Performance Comparison of Various Pitch Predictor Schemes for 4.8 kbps

The hybrid adaptation with a pitch tracker is very sensitive to channel errors and does not contribute largely to the performance of the tree coder. The use of the smoothed input in the pitch predictor gives a notable error performance increase from 3 dB to 6 dB over Cuperman's hybrid pitch predictor at 10^{-3} BER. The pitch predictor with a variable smoother provides an increase of about 0.6 dB in the average SNRSEG of five sentences over the pitch predictor without a smoother in the noise free channel. The performance improvement is notable in the low pitched male speech. The pitch predictor with a variable smoother can track the pitch period change in a block because the coefficients of the smoother are assigned depending on the change of pitch period. Also it partly reduces the mismatch between the real pitch period and its representation by integer multiples of the sampling interval due to a sampling rate reduction.

Ⅳ. Tree coder combined with TDHS at 4.8 kbps

The overall coding system combining tree coding with TDHS is shown in Fig. 6. The TDHS algorithm is used to compress the bandwidth of the input speech by a factor of two. The pitch period is extracted from a block of speech samples using the AMDF method. The compressed speech is encoded by a tree coder. The output of the tree coder is a string of path map symbols. At the receiver, the compressed speech signal is reconstructed by a tree decoder and expanded by the TDHS expansion operation.

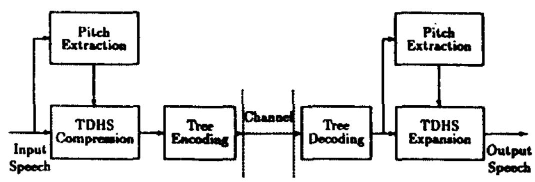
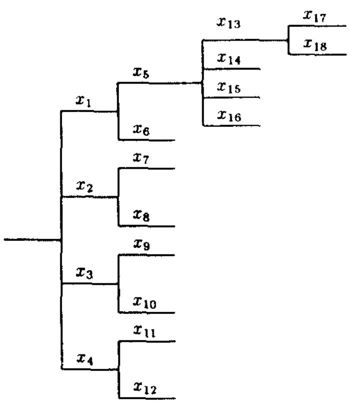


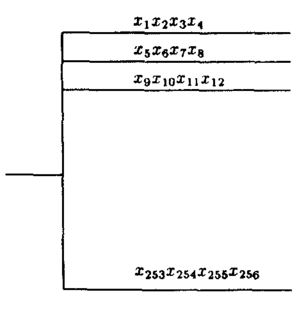
Fig. 6. System Configuration for Combining TDHS and Tree Coding

A. Different Code Trees

For the tree coding combined with TDHS at 4.8 kbps, 1.5 bits/sample can be assigned to the tree coder at a sampling rate of 6.4 kHz. The fractional rate tree coders can be designed according to the number of branches per node and the number of symbols per branch. A multitree coder and a vector tree coder are examined here. The 1.5 bits/sample multitree coder consists of a tree with 4 branches and a tree with 2 branches, which are interleaved with each other. Fig. 7(a) shows the 1.5 bits/sample multitree code. The output symbols on the branches of the multitree are assigned as the output values of the MMSE Gaussian quantizer, both scaled by the gain term. The gain is adapted by using the hybrid adaptive quantizer in a 4 level tree and delta modulator adaptation in a 2 level tree.



(a) 4-2 Multitree Coder



(b) Vector Tree Coder

Fig. 7. Code Trees for 1.5 Bits/sample

A Gaussian stochastic code tree at 4.8 kbps is examined. The vector code tree has 64 branches per node and 4 symbols per branch. The branch labels in the code tree are taken from a stochastic codebook populated with Gaussian random variates. The gain term in the code generator is calculated using the modified vector gain algorithm. The code generator of both coders consists of a pole zero short term predictor and a pitch predictor.

Table I compares the SNR/SNRSEG performance of two code trees in the tree coding part and TDSH/tree coder. The stochastic vector tree may be somewhat better than the 4 2 multitree through various speech data in the error free channel, but the error performance of the stochastic vector tree degrades more rapidly than 4 2 multitree as the BER increases. One bit error of the index in the stochastic vector tree affects the whole vector with 4 samples. The 4-2 multitree coder achieves almost equivalent performance to the stochastic vector tree coder in the noiseless channel even though it is less complex. Subjectively, the tree coding combined with TDHS at the rate of 4.8 kbps shows a good quality of speech and is roughly close to a μ law PCM system with 5 bit to 6 bit. The narrowband spectrograms of the original speech and the reconstructed output for 4 2 multitree coder combined with TDHS at the rate of 4.8 kbps are shown in Fig. 8 and Fig. 9. The spectrogram of the 4.8 kbps coder maintains the original spectrogram relatively well.

Table I. Performance Comparison of Two Code Trees

		Tree Coder: SNR/SNRSEG, Overall: SNR/SNRSEG			
		4-2 Multitree Coder		Vector Tree Coder	
Sent.	BER	Tree Coder	Overall	Tree Coder	Overall
Fem.	0	17.10/16.72	14.12/12.24	17.07/16.41	14.17/12.10
	10^{-4}	16.54/16.29	13.82/12.01	15.88/15.69	13.48/11.67
	10^{-3}	12.82/13.00	11.57/ 9.93	9.11/10.88	8.64/ 8.40
	10^{-2}	6.01/ 5.77	6.06/ 4.91	-0.09/ 1.16	0.26/ 0.85
Male	0	12.42/14.87	10.55/ 9.89	13.31/15.33	11.11/10.21
	10^{-4}	11.89/14.25	10.21/ 9.56	12.04/14.29	10.31/ 9.63
	10^{-3}	9.57/10.30	8.72/ 7.48	6.79/ 9.04	6.62/ 6.63
	10^{-2}	3.87/ 3.83	4.01/ 3.04	-0.97/-0.52	-0.36/-0.46

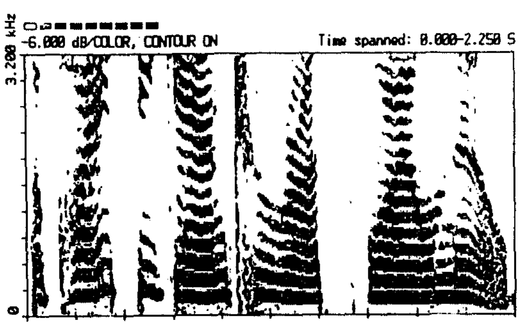


Fig. 8. Spectrogram of Original Speech

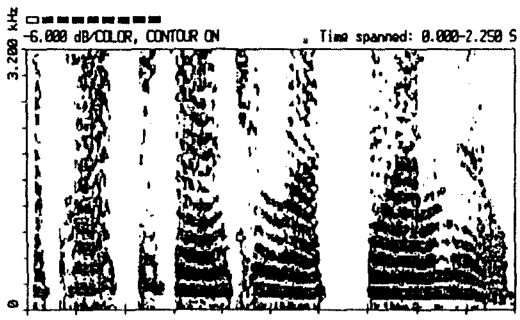


Fig. 9. Spectrogram of Reconstructed Speech in Noiseless Channel for 4.8 kbps TDHS-TREE Coder

Ⅴ. Conclusions

Efficient speech coders using tree coding combined with harmonic scaling were designed at the rate of 4.8 kbps. A time domain harmonic scaling algorithm (TDHS) was used to compress input speech by a factor of two. This process allowed the tree coder to have 1.5 bits/sample for 4.8 kbps in the case of a 6.4 kHz sampling rate. The tree coding was designed with an emphasis on robustness to transmission errors in the digital channel as well as good quality of speech in the error free channel. The performance of a tree coder was enhanced by improving the code generator. In a backward adaptive tree coder, there are three components of the predictor in the code generator : a pitch predictor, a short-term all-zero predictor, and a short-term all-pole predictor. These three predictors are connected in cascade in the code generator. The robustness of the tree coder could be improved by careful choosing the input of the short-term predictor adaptation and including a smoother in the pitch predictor.

A 4-2 multitree coder and a vector tree coder with a stochastic codebook with a size of 64 were investigated. Subjectively, TDHS-tree coding achieved a good quality of speech at a rate of 4.8 kbps. The speech quality of tree coding combined with TDHS at a rate of 4.8 kbps was close to that of 1.5 bits/sample tree coding of uncompressed speech. Also, it contained little perceived distortion at a BER of 10^{-4} and only small perceived distortion at 10^{-3}. For a BER of 10^{-2}, the reconstructed speech was intelligible and easily understandable.

REFERENCES

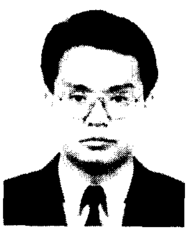
1. D. Malah, "Time-domain algorithms for harmonic bandwidth reduction and time scaling of speech signals," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-27, pp. 121-133, Apr. 1979.
2. D. Malah, "Combined time-domain harmonic compression and CVSD for 7.2 kbits/sec transmission of speech signals," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, pp. 504-507, Apr. 1980.
3. D. Malah, R. E. Crochiere, and R. V. Cox, "Performance of transform and subband coding systems combined with harmonic scaling of speech," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-29, pp. 273-283, 1981.
4. J. B. Anderson and J. B. Bodie, "Tree encoding of speech," *IEEE Trans. Inform. Theory*, vol. IT-21, pp. 379-387, July 1975.
5. J. D. Gibson and W. W. Chang, "Fractional rate multitree speech coding," *IEEE Trans. Commun.*, vol. 39, pp. 963-974, June 1991.
6. I. Lee and J. D. Gibson, "Tree coding combined with harmonic scaling of speech at 6.4 kbps," in *Proc. IEEE Conf. Acoust. Speech, Signal Processing*, pp. 600-603, Apr. 1993.

7. H. C. Woo and J. D. Gibson, "Low delay tree coding of speech at 8 kbps," in *Conf. Rec. IEEE Global Telecommun. Conf.*, (Phoenix, AZ), pp. 1884-1888, Dec. 1991.
8. R. C. Reininger and J. D. Gibson, "Bakward adaptive lattice and transversal predictors in ADPCM," *IEEE Trans. Commun.*, vol. COM-33, pp. 74-82, Jan. 1985.
9. Y. Miyanaga, N. Miki, N. Nagai, and K. Hatori, "A speech analysis algorithm which eliminates the influence of pitch using the model reference adaptive system," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-30, pp. 88-96, Feb 1982.
10. N. S. Jayant and P. Noll, *Digital Coding of Waveforms: Principles and Applications to Speech and Video*. Englewood Cliffs, NJ: Prentice-Hall, 1984.
11. J. Durbin, "Efficient estimation of parameters in moving average models," *Biometrika*, vol. 46, pp. 306-316, 1959.
12. R. Pettigrew and V. Cuperman, "Backward pitch prediction for low delay speech coder," in *Conf. Rec. IEEE Global Telecommun. Conf.*, pp. 1247-1252, 1989.

姜 尙 遠(Sangwon Kang) 정회원

1980년 2월 : 한양대학교 전자공학과 졸업(공학사)
1982년 2월 : 서울대학교 전자공학과 대학원 졸업(공학석사)
1990년 8월 : 미국 Texas A&M Univ. 전기공학과 졸업(공학박사)
1982년 12월 ~ 현재 : 한국전자통신연구소 이동통신연구단 신호처리연구실장
※주관심분야 : 이동통신, 음성신호처리, 통신이론

李 寅 誠(Insung Lee) 정회원
1960년 6월 8일 생
1983년 2월 : 연세대학교 전자공학과 졸업(공학사)
1985년 2월 : 연세대학교 전자공학과 대학원 졸업(공학석사)
1992년 12월 : 미국 Texas A&M Univ. 전기공학과 졸업(Ph.D.)
1986년 5월 ~ 1987년 7월 : 한국통신 연구개발단 전임연구원
1993년 2월 ~ 현재 : 한국전자통신연구소 신호처리연구실 선임연구원
※주관심분야 : 음성, 영상데이타 압축, 적응필터, 이동통신



韓 敬 浩(Kyongho Han) 정회원
1959년 6월 25일 생
1982년 2월 : 서울대학교 전자공학과 졸업(공학사)
1984년 2월 : 서울대학교 전자공학과 대학원 졸업(공학석사)
1992년 8월 : 미국 Texas A&M Univ. 전기공학과 졸업(Ph.D.)
1985년 5월 ~ 1987년 12월 : 한국전기통신공사 품질보증단 선임연구원
1992년 10월 ~ 1993년 8월 : 한국전자통신연구소 신호처리연구실 선임연구원
1993년 8월 ~ 현재 : 단국대학교(서울) 전기공학과
※주관심분야 : 컴퓨터구조, VLSI, 마이크로프로세서응용.