Tandeming Analysis of Perceptual Pre-weighting and Post-weighting Multimode Tree Coder

Ying-Yi Li, Pravin Ramadas, and Jerry D. Gibson
Department of Electrical and Computer Engineering,
University of California, Santa Barbara, USA
Email: {yingyi_li, pravin_ramadas}@umail.ucsb.edu, gibson@ece.ucsb.edu

***Abstract*—The perceptual pre-weighting and post-weighting Multimode Tree Coder is low delay and low complexity. Since the tandem connection of different codecs in voice calls is common today, it is also important to assess any loss in end-to-end speech quality caused by asynchronous tandem coding. We evaluate the tandeming performance of our Multimode Tree Coder when tandemed with itself, with G.727, and with the AMR-NB codec. The results show that the tandem performance of the Multimode Tree Coder is comparable to the AMR-NB coder at 12.2 kbps.**

I. Introduction

A low delay, low complexity, and low bit-rate speech coder would be attractive for Voice over IP (VoIP) and Voice over Wireless LAN (VoWLAN) applications. To address these applications, we have proposed a phonetically switched Multimode Tree Coder (MMT) with the G.727 backward adaptive code generator that exhibits these characteristics [1]. Although it is not well known, the tandem connection of different codecs in voice calls is common today. For example, a mobile to mobile digital cellular call connected through a wireline VoIP connection often involves 3 different speech codecs, a different codec for each mobile and a different codec in the VoIP backbone. The coded speech thus has to be transcoded (decoded and re-encoded) at each network interface. These transcoding operations between codecs, called asynchronous tandeming of codecs, results in increased latency as well as performance degradation. A low delay codec helps to reduce the delay but it is important to assess any loss in end-to-end speech quality caused by asynchronous tandem coding.

The Multimode Tree Coder is based on Multimode classification and Tree coding [1], [2]. Multimode coding is based on phonetic classification of speech. The speech is classified into five modes and each mode is coded with a suitable bit-rate. Tree coding is an encoding procedure where speech samples are coded effectively based on the best long term tree-structured fit to the input waveform [3], [4]. In order to reduce the computational complexity of the perceptual distortion calculation in the Tree Search, we introduced pre-weighting and post-weighting filters in our Multimode Tree Coder in [1].

Since the Multimode Tree Coder [1] is low delay and low complexity, it helps to reduce the delay of transcoding operations. However, the speech quality of asynchronous tandem coding is also an important issue. Therefore, we compare the tandeming performance of the Multimode Tree Coder with G.727 and AMR-NB. The tandeming performance is evaluated by PESQ [5], an objective method for end-to-end speech quality assessment of narrow-band telephone networks and speech codecs. The results show that the tandeming performance of the Multimode Tree Coder is comparable to AMR-NB 12.2 kbps codec for both clean and noisy sequences.

The paper is organized as follows. Section II describes tree coding basics. Section III discusses the details of the Multimode Tree Coder with perceptual pre-weighting and post-weighting. The tandeming performance of the speech codecs is compared in Section IV. Finally, conclusions are presented in Section V.

This research has been supported by NSF Grant Nos. CCF-0728646 and CCF-0917230.

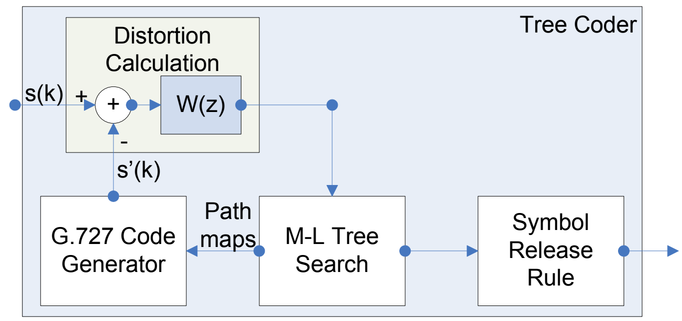

Fig. 1. Tree Coder without pre- and post-weighting

II. Tree Coding

A Tree coder has a Code Generator, a Tree Search algorithm, a distortion measure and a path map symbol release rule as shown in Fig. 1. The Tree Search algorithm, in combination with the Code Generator and appropriate distortion measure, chooses the best candidate path to encode the current input sample. The symbol release rule decides the symbols on the best path to encode. For simplicity, we used G.727 as our Code Generator since it is a low delay and low complexity ADPCM coder. The coding bit-rate is controlled by the result of the mode decision. In order to reduce the computational complexity, we used the *M-L* Tree Search as Tree Search algorithm. The *M* paths with minimum cumulative distortion are chosen and extended along their children. The distortion of each path is calculated with a perceptual weighting filter, which helps to choose a path where the noise is masked by

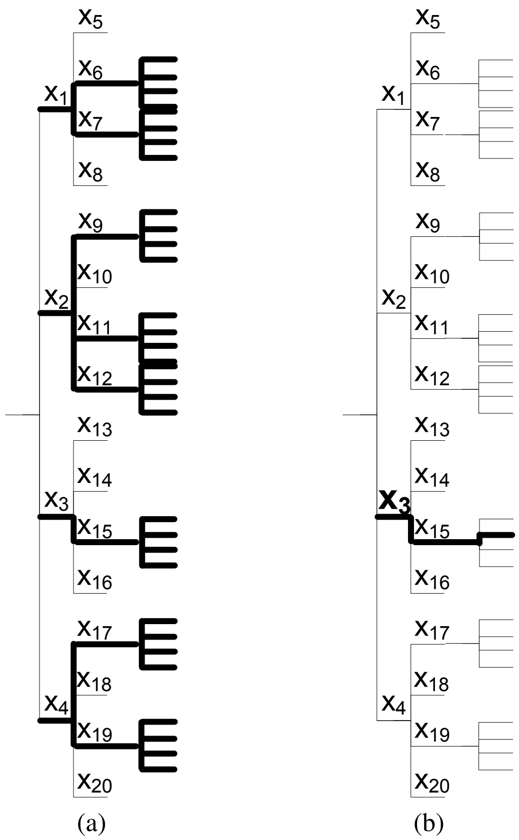

Fig. 2. An example of 16 kbps tree (a) Search paths of *M-L* Tree Search for $L = 3$ and $M = 8$ (b) Symbol release rule

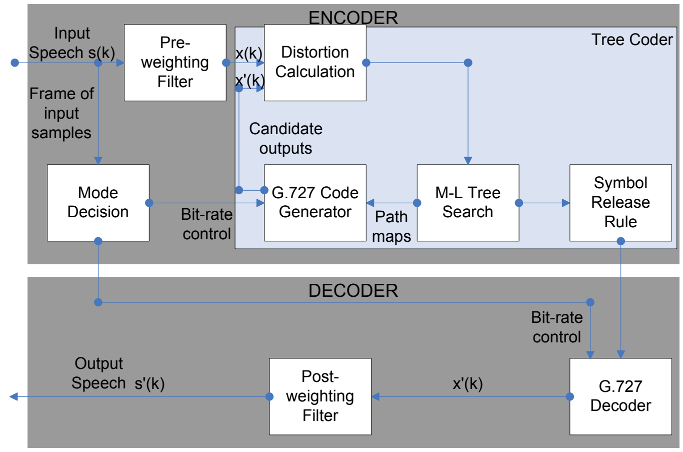

Fig. 3. Multimode Tree Coder with pre-weighting and post-weighting

the speech spectrum. The weighting filter is

$$W(z) = \frac{1 - \sum_{i=1}^{N} a_i z^{-i}}{1 - \sum_{i=1}^{N} \mu^i a_i z^{-i}}, \quad (1)$$

where μ is 0.86, N is 5, and a_i's are the short term predictor coefficients calculated from the current speech frame. Finally, the symbol corresponding to the first node in the minimum cumulative distortion path is transmitted.

For example, there are 4^L paths of a tree generated with a 16 kbps ADPCM coder as shown in Fig. 2. Assume $L = 3$ and $M = 8$, the 8 minimum cumulative distortion paths, $x_1 \rightarrow x_6$, $x_1 \rightarrow x_7$, $x_2 \rightarrow x_9$, $x_2 \rightarrow x_{11}$, $x_2 \rightarrow x_{12}$, $x_3 \rightarrow x_{15}$, $x_4 \rightarrow x_{17}$, and $x_4 \rightarrow x_{18}$, with their children are marked as search paths in Fig. 2 (a). Based on *M-L* Tree Search, we only need to maintain M, minimum cumulative distortion paths instead of 4^L paths, which saves the computational complexity for tree searching. In Fig. 2 (b), the minimum cumulative distortion path, $x_3 \rightarrow x_{15}$, is marked. By the single symbol release rule, the symbol x_3 is released and encoded.

III. Perceptual Pre-weighting and Post-weighting Multimode Tree Coder

The Multimode Tree Coder with perceptual pre- and post-weighting contains three main parts, mode decision, Tree coder, and pre- and post-weighting filters. The block diagram of the Multimode Tree Coder with perceptual pre- and post-weighting is shown in Fig. 3. In a Multimode Tree Coder, the bit-rate of coding each input sample is controlled by the mode decision of the current frame. The input speech is filtered by the Pre-weighting filter in the encoder. Based on the results of the mode decision, the Code Generator codes the pre-weighted sample at a suitable bit-rate. Then the distortion between candidate outputs and pre-weighted input samples is calculated via the *M-L* Tree Search, which is a tree search with depth L and M retained paths to depth L. Finally, the first symbol in the minimum distortion path is released. In the decoder, a G.727 decoder is used to decode the coded symbol since we use G.727 [6] as our Code Generator in our tree coder. Because of the pre-weighting in the encoder, the post-weighting filter is required in the decoder. The decoded sequence from the G.727 decoder is filtered by the post-weighting filter, and the reconstructed signal is produced.

A. Mode Decision

The mode decision of the Multimode Tree Coder is a low delay and low complexity method based on G.727 ADPCM coder state parameters, step-size scale factor and long-term average magnitude of weighted quantization level, and frame energy. A speech frame of 40 samples is classified into one of these five modes: Voiced (V), Onset (ON), Unvoiced (UV), Hangover (H), and Silence (S). Each mode is coded at a suitable bit-rate. We used 24 kbps for Voiced (V) and Onset (ON), and 16 kbps for Unvoiced (UV) and Hangover(H). Silence is coded by comfort noise coding, and the average bit-rate of silence is 0.72 kbps.

B. Perceptual Pre-weighting and Post-weighting

The complexity of the speech codec should be as low as possible. However, the computational complexity with the perceptual weighting filter inside the loop as in Fig. 1 is high. Assume that the computational complexity of $W(z)$ is C operations, and B is the number of siblings of the tree such as $B = 4$ for the 16 kbps tree, then the complexity of releasing one symbol is $M \cdot B \cdot L \cdot C$ operations. Schuller, Yu, Huang, and Edler [7] have employed adaptive pre-filtering and post-filtering in lossless audio coding. They showed that lossless audio coding with pre- and post-filtering still

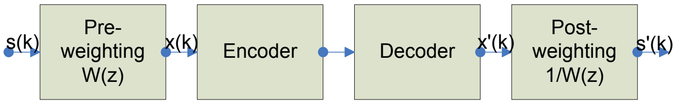

Fig. 4. Pre-weighting and post-weighting filters for speech codecs

keeps the high quality. In addition, Shetty and Gibson [8] employed perceptual pre-weighting and post-weighting in a G.726 ADPCM codec and a modified AMR-NB CELP codec. They showed that the performance of lossy coding with pre- and post-weighting also performs well. As shown in Fig. 4, the computational complexity of our Multimode Tree Coder is reduced to $2C$ operations for releasing one symbol by using pre-weighting and post-weighting filters.

The design of the pre-weighting filter $W(z)$ and post-weighting filter $\frac{1}{W(z)}$ is to mask the reconstruction error at the output by the input spectrum. Let $S(z)$ be the input speech, $X(z)$ be the pre-weighted speech, $X'(z)$ be the pre-weighted speech output, and $S'(z)$ be the output speech after post-weighting. From Fig. 4, the relation of $S(z)$ and $X(z)$ is

$$S(z)W(z) = X(z), \tag{2}$$

and the relation of $S'(z)$ and $X'(z)$ is

$$X'(z)\frac{1}{W(z)} = S'(z). \tag{3}$$

Let $E(z)$ denote the coding error for the pre-weighted speech. From Eq. (2) and Eq. (3), the coding error $E(z)$ will be

$$\begin{aligned} E(z) &= X(z) - X'(z) \\ &= W(z)S(z) - W(z)S'(z) \\ &= W(z)[S(z) - S'(z)], \end{aligned} \tag{4}$$

where $W(z)$ is used to shape the reconstruction error [8].

The objective is to match the frequency response of perceptual error weighting filter generated with 5th order LPC coefficients in Eq. (1) with the frequency response of the filter generated with ADPCM predictor coefficients. The post-weighting filter of 5th order LPC coefficients is

$$\frac{1}{W(z)} = \frac{1 - \sum_{i=1}^{5} (0.86)^i a_i z^{-i}}{1 - \sum_{i=1}^{5} a_i z^{-i}}, \tag{5}$$

while the post-weighting filter generated with ADPCM pole-zero coefficients is

$$H_{post}(z) = \frac{1 + \sum_{i=1}^{6} m_2^i b_i z^{-i}}{(1 + \sum_{i=1}^{6} m_3^i b_i z^{-i})(1 - \sum_{i=1}^{2} m_1^i a_i z^{-i})}, \tag{6}$$

where a_i's are pole coefficients, b_i's are zero coefficients, $m_1 = 0.2$, $m_2 = 1.0$, and $m_3 = 0.85$ in both pre- and post-weighting filters in our experiments.

TABLE I
THE NARROWBAND CLEAN TEST SEQUENCES (8000 SAMPLES/SEC)

Sequence	Language	Male/Female
T04	Spanish	Female
T05	Spanish	Male
T06	English	Female
T07	English	Female
T08	English	Female
T12	English	Male
T13	English	Male

TABLE II
THE NARROWBAND NOISY TEST SEQUENCES (8000 SAMPLES/SEC)

Sequence	Language	Male/Female	SNR
F1	English	Male	10 dB Car noise
F2	English	Male	15 dB Car noise
F3	English	Male	20 dB Car noise
L1	English	Female	10 dB Train noise
L2	English	Female	15 dB Train noise
L3	English	Female	20 dB Train noise
W1	English	Female	10 dB Airport noise
W2	English	Female	15 dB Airport noise
W3	English	Female	20 dB Airport noise

IV. RESULTS

In our experiments, the depth of tree, *L*, is 10, and *M* is 4 in our current codec for the *M-L* Tree Search. The Voiced (V) and Onset (ON) modes are coded at 24 kbps while Unvoiced (UV) and Hangover (H) modes are coded at 16 kbps. PESQ [5] is used for evaluating the speech quality of the narrowband coder. We compare the tandeming performance of the Multimode Tree Coder to the AMR-NB coder at 12.2 kbps [9] and G.727 ADPCM coding at 32 kbps with 4 core bits. The source controlled rate operation of AMR-NB is enabled. 7 clean sequences and 9 noisy sequences are used. The evaluation of the performance for noisy sequences uses the clean version of the noisy sequence as the reference and the decoded noisy sequence as the degraded sequence. The details of the test sequences [10] are listed in Tables I and II.

A. Effects of Perceptual Error Weighting Filter

Since the perceptual error weighting filter masks the noise by the speech spectrum, the performance of the Multimode tree coder with the perceptual error weighting filter for noisy sequences should be better than that without perceptual weighting. Table III shows the results of the Multimode Tree Coder with/without perceptual error weighting filter for

TABLE III
THE PESQ WITH/WITHOUT PERCEPTUAL WEIGHTING FOR NOISY SEQUENCES

Sequence	w/ weighting	w/o weighting	uncoded
F1	2.447	2.316	2.338
F2	2.707	2.573	2.612
F3	2.973	2.840	2.871
L1	2.719	2.647	2.712
L2	3.036	2.942	3.043
L3	3.265	3.193	3.331
W1	2.520	2.430	2.496
W2	2.720	2.644	2.727
W3	3.018	2.907	2.940
Average	2.823	2.721	2.786

TABLE IV
THE PESQ OF SELF-TANDEM AND CROSS-TANDEM OF MMT AND AMR-NB FOR CLEAN SEQUENCES

Sequence	MMT–MMT	AMR–AMR	MMT–AMR	AMR–MMT
T04	3.523	3.579	3.641	3.551
T05	3.769	3.846	3.800	3.662
T06	3.543	3.664	3.608	3.474
T07	3.568	3.524	3.605	3.463
T08	3.728	3.781	3.833	3.618
T12	3.508	3.709	3.630	3.560
T13	3.609	3.901	3.840	3.505
Average	3.607	3.715	3.708	3.548

TABLE V
THE AVERAGE BIT-RATE (KBPS) OF 3 SPEECH CODECS FOR CLEAN SEQUENCES

Sequence	MMT	AMR	G.727
T04	21.62	10.62	32
T05	20.74	10.53	32
T06	17.52	7.37	32
T07	18.50	8.47	32
T08	20.52	9.17	32
T12	16.67	7.68	32
T13	19.06	8.34	32
Average	19.23	8.88	32

noisy sequences. The results show that the perceptual error weighting filter does improve the perceptual speech quality since it masks the noise spectrum.

B. Self-Tandem and Cross-Tandem of MMT and AMR-NB

Self-tandem and cross-tandem performance of the Multimode Tree Coder and the AMR-NB coder are compared. Table IV shows the results of the self-tandem and cross-tandem performance of our Multimode Tree Coder and the AMR-NB coder for clean sequences. The average bit-rate of each speech codec for clean sequences is shown in Table V. The self- and cross-tandem performance for noisy sequences are shown in Table VI, and Table VII shows the average bit-rate of each speech codecs. In the tables, MMT stands for our Multimode Tree Coder, AMR stands for AMR-NB coder at 12.2 kbps, and G.727 stands for the G.727 coder at 32 kbps. For simplification, the order of tandeming is expressed as X-Y, where X is the first stage coder while Y is the second stage coder.

TABLE VI
THE PESQ OF SELF-TANDEM AND CROSS-TANDEM OF MMT AND AMR-NB FOR NOISY SEQUENCES

Sequence	MMT–MMT	AMR–AMR	MMT–AMR	AMR–MMT
F1	2.535	2.560	2.571	2.564
F2	2.780	2.808	2.802	2.784
F3	3.048	3.080	3.078	3.055
L1	2.710	2.876	2.838	2.722
L2	2.950	3.175	3.121	3.077
L3	3.188	3.376	3.265	3.307
W1	2.564	2.595	2.595	2.557
W2	2.771	2.808	2.787	2.750
W3	2.990	3.017	3.066	2.968
Average	2.837	2.922	2.903	2.865

TABLE VII
THE AVERAGE BIT-RATE (KBPS) OF 3 SPEECH CODECS FOR NOISY SEQUENCES

Sequence	MMT	AMR	G.727
F1	21.91	12.20	32
F2	21.95	12.14	32
F3	21.35	9.86	32
L1	22.31	12.20	32
L2	22.13	11.72	32
L3	19.60	11.42	32
W1	20.20	12.20	32
W2	20.36	12.04	32
W3	21.40	10.42	32
Average	21.25	11.58	32

From Tables IV and VI, the self-tandem performance of the Multimode Tree Coder is similar to that of the AMR-NB coder for both clean and noisy sequences. Moreover, the cross-tandem performance of the two codecs is also comparable, and the tandeming results are better than original noisy uncoded sequences as shown in Table III. This is because both Multimode Tree Coder and AMR-NB use perceptual error weighting filter, and the noise is masked by the speech spectrum as shown in Section IV-A. Therefore, the performance of noisy coded sequences is improved. Based on our experimental results, the tandem performance of our Multimode Tree Coder is similar to that of the AMR-NB coder for both cross-tandem and self-tandem performance for all sequences, and there is no audible degradation of the speech quality.

TABLE VIII
THE PESQ OF MMT AND AMR-NB TANDEM WITH G.727 FOR CLEAN SEQUENCES

Sequence	G.727–MMT	G.727–AMR	MMT–G.727	AMR–G.727	G.727–G.727
T04	3.702	3.734	3.412	3.528	3.771
T05	3.732	3.915	3.688	3.515	3.901
T06	3.689	3.701	3.553	3.591	4.002
T07	3.779	3.749	3.637	3.535	4.007
T08	3.867	3.980	3.823	3.810	4.087
T12	3.720	3.673	3.648	3.616	3.819
T13	3.713	3.919	3.672	3.647	3.852
Average	3.743	3.810	3.633	3.606	3.920

TABLE IX
THE PESQ OF MMT AND AMR-NB TANDEM WITH G.727 FOR NOISY SEQUENCES

Sequence	G.727–MMT	G.727–AMR	MMT–G.727	AMR–G.727	G.727–G.727
F1	2.434	2.480	2.439	2.463	2.318
F2	2.691	2.734	2.694	2.714	2.587
F3	2.961	2.982	2.955	2.960	2.852
L1	2.706	2.785	2.697	2.799	2.677
L2	2.996	3.101	2.987	3.102	2.989
L3	3.247	3.348	3.226	3.381	3.290
W1	2.511	2.575	2.509	2.556	2.488
W2	2.714	2.789	2.706	2.779	2.717
W3	3.010	3.015	2.999	3.006	2.930
Average	2.808	2.868	2.801	2.862	2.761

C. Cross-Tandem Performance with G.727

The cross-tandem performance of the Multimode Tree Coder and AMR-NB when tandemed with G.727 are compared. The results of clean sequences are shown in Table VIII.

TABLE X
COMPARISON OF MMT, AMR-NB, AND G.727

Speech Coder	MMT		AMR-NB		G.727	
Attributes	clean	noisy	clean	noisy	clean	noisy
PESQ	3.8-4.0	2.4-3.3	3.7-4.1	2.4-3.4	3.8-4.1	2.3-3.3
Avg Bit-rate (kbps)	16.67-21.62	19.60-22.31	7.37-10.62	10.42-12.20	32	32
Algorithmic Delay (ms)	6.125		25		0.125	
Comp. Complexity (WMOPS)	3.3-6.1 [1]		11.9-16.7 [11]		1.50 [12]	

Table IX has the results of the noisy sequences. From Table VIII, because of comfort noise coding in the Multimode Tree Coder and the AMR-NB coder, their tandeming performance is worse than that of G.727 coder. However, the tandeming performance of our Multimode Tree Coder is similar to that of the AMR-NB coder.

In Table IX, we see the tandeming results of our Multimode Tree Coder and AMR-NB coder are better than those of G.727 coder for noisy sequences. This is because we use a perceptual error weighting filter and VAD. In addition, the tandeming performance of the Multimode Tree Coder is also comparable to that of AMR-NB. Moreover, Table VIII and Table IX also show that there is no significant audible degradation between the Multimode Tree Coder and the G.727 coder.

D. Analysis of the Multimode Tree Coder

The Multimode Tree Coder combines the principle of Multimode classification with Tree coding. Multimode coding reduces the average bit-rate of the coder. Even though Tree Coding is a delayed coding, the code generator is backward adaptive, so there is no frame delay and the look-ahead delay is low. Since G.727 ADPCM is a low complexity coder, the Tree coder with a G.727 Code Generator has low computational complexity. Moreover, the *M-L* Tree Search saves the computational complexity of the Tree Search, and the distortion calculation using pre-weighting and post-weighting is also very computationally efficient. The comparisons of computational complexity are shown in Table X.

We already showed that the tandeming performance of our Multimode Tree Coder is comparable to the AMR-NB at 12.2 kbps in the previous sections. Even though the bit-rate of the Multimode Tree Coder is twice that of AMR-NB, it saves about one-third the bit-rate of G.727 for both clean and noisy sequences. However, the delay and the computational complexity of our Multimode Tree coder are much lower than those of AMR-NB. From Table X, the delay of the Multimode Tree Coder is about a quarter of AMR-NB, and the worst complexity is about one-third of AMR-NB.

V. CONCLUSIONS

The tandem connection of different codecs in voice calls is common today. The coded speech has to be transcoded at each network interface, and the asynchronous tandeming of codecs results in increased latency and performance degradation. Even though a low delay speech codec helps to reduce the delay, it is important to assess any loss in end-to-end speech quality caused by asynchronous tandem coding. Therefore, we assess the tandeming performance of speech codecs.

In this paper, we evaluate the tandeming performance of our Multimode Tree Coder when tandemed with itself, with G.727, and with the AMR-NB codec often used in digital cellular and also in some VoIP applications. The tandeming performance of the Multimode Tree Coder with a G.727 Code Generator is comparable to that of AMR-NB at 12.2 kbps. There is no significant audible degradation for both clean and noisy sequences. In addition, the tandeming performance of noisy sequences is improved due to the perceptual error weighting filter. Thus, in addition to low delay and low complexity, good tandeming performance is also achieved by the Multimode Tree Coder [1].

REFERENCES

[1] P. Ramadas, Y.-Y. Li, and J. D. Gibson, "Multimode Tree Coding of Speech with Perceptual Pre-weighting and Post-weighting," *the 12th IASTED International Conference on Signal and Image Processing*, 2010.

[2] P. Ramadas and J. D. Gibson, "Phonetically Switched Tree coding of speech with a G.727 Code Generator," *the 43rd Annual Asilomar Conference on Signals, Systems, and Computers*, Nov. 1-4, 2009.

[3] A. Goris and J. D. Gibson, "Incremental tree coding of speech," *IEEE Trans. on Information Theory*, vol. 27, no. 4, pp. 511 – 516, July 1981.

[4] H. C. Woo and J. D. Gibson, "Low delay tree coding of speech at 8 kbit/s," *IEEE Trans. on Speech and Audio Processing*, vol. 2, no. 3, pp. 361 –370, July 1994.

[5] ITU-T Recommendation P.862, "Perceptual Evaluation of Speech Quality (PESQ), an objective method for end-to-end Speech Quality Assessment of Narrow-band telephone networks and Speech Codecs," Feb. 2001.

[6] ITU-T Recommendation G.727, "5-, 4-, 3- and 2-bit/sample embedded Adaptive Differential Pulse Code Modulation (ADPCM)," Dec. 1990.

[7] G. Schuller, B. Yu, D. Huang, and B. Edler, "Perceptual Audio Coding using Adaptive Pre- and Post-Filters and Lossless Compression," *IEEE Trans. on Speech and Audio Processing*, vol. 10, no. 6, pp. 379 –390, Sept. 2002.

[8] N. Shetty and J. D. Gibson, "Perceptual Pre-weighting and Post-inverse weighting for Speech Coding," *the 41st Annual Asilomar Conference on Signals, Systems, and Computers*, Nov. 4-7, 2007.

[9] 3GPP, "AMR speech Codec; C-source code," 3rd Generation Partnership Project (3GPP), TS 26.073, June 2007. [Online]. Available: http://www.3gpp.org/ftp/Specs/html-info/26073.htm

[10] ——, "AMR speech Codec; Test sequences," 3rd Generation Partnership Project (3GPP), TS 26.074, June 2007. [Online]. Available: http://www.3gpp.org/ftp/Specs/html-info/26074.htm

[11] 3GPP TSG-S4, "AMR Wideband Permanent project document WB-4: Design Constraints, v. 1.0," Mar. 2000.

[12] J. D. Gibson, "Speech Coding Methods, Standards, and Applications," *IEEE Circuits and Systems Magazine*, vol. 5, no. 4, pp. 30 – 49, Fouth Quarter 2005.