A TWO STAGE HYBRID EMBEDDED SPEECH/AUDIO CODING STRUCTURE

Sean A. Ramprashad

Bell Laboratories, Lucent Technologies
700 Mountain Ave, Murray Hill, NJ 07974
ramprash@research.bell-labs.com

ABSTRACT

A two stage hybrid embedded speech/audio coding structure is proposed. The structure uses a speech coder as a core to provide the minimal bitrate and an acceptable performance on speech inputs. The second stage is transform coder using a MDCT and perceptual coding principles. This stage is itself embedded both in complexity and bitrate, and provides various levels of enhancement of the core output, particularly for general audio signals like music. Informal A-B comparison tests show that the performance of the structure at 16 kb/s is between that of the GSM Enhanced Full Rate coder at 12.2 kb/s, and the G.728 LD-CELP coder at 16 kb/s.

1. INTRODUCTION

Traditionally speech and audio coders have been designed with a single application in mind. For example, low bitrate speech coders provide high compression ratios and low algorithmic delays needed in applications such as wireless communications. These coders tend not to work well on background noise and general audio signals such as music. Audio coders operate at higher bitrates and delays, and can be rather complex. They are suitable for storage and broadcast applications, or for communication on networks where bandwidth and processing power are not as restrictive.

Advances in the integration of many networks, wireless, data, voice, and others, and the mobility of users, has expanded the range that speech/audio coders will operate. It is now possible for a call to originate from one network, say a wireless network in a car, and terminate on a completely different network like an IP network in an office.

The implication is that future coders should have the ability to adapt and operate simultaneously under multiple constraints of bitrate, complexity, delay, and robustness to input signal. In addition, many communication systems have already adopted and deployed more traditional coding standards. How to enhance these existing standards to provide added levels of performance and flexibility without modifying the deployed algorithm is a challenge.

This paper addresses both of these concerns. Proposed is a two stage coding structure with a speech coder at the core. The core by itself has the lowest bitrate, delay, and complexity, providing a minimum performance useful for interactive speech communication. Two coders, G.723.1 and G.729, are used in this paper for the core [1] [2]. The second stage is a transform coder which is linked to the first stage only through the output of the first stage (core) decoder. This second stage is itself embedded, and its bitrate and complexity can be pruned both during and after encoding to provide various levels of performance.

The paper is outlined as follows. Section 2 describes the two stage structure, pointing out various advantages and disadvantages of the architecture. Section 3 provides the motivation for using a hybrid structure, and Section 4 discusses the transform used. Section 5 gives a brief overview of the algorithm and bitstream, and Section 6 gives results of informal A-B comparison tests. Finally, Section 7 provides some closing remarks.

2. THE TWO STAGE STRUCTURE

The general two stage structure is shown in Figure 1.

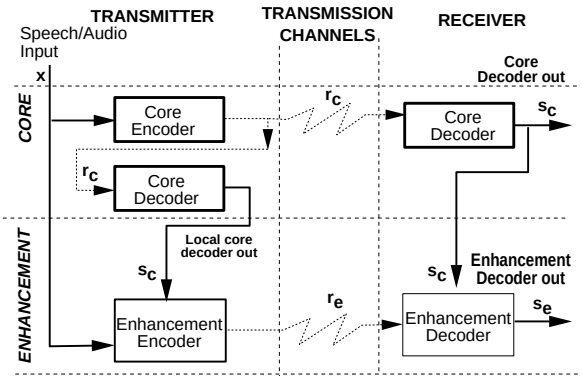

Figure 1: Two Stage Embedded Structure.

The enhancement stage depends *minimally* on the core stages only through the core decoder output. This is an important difference when compared to other embedded structures [3] [4] [5] [6], having both advantages and disadvantages. The first advantage is that the second stage can be linked to different core stages with little or no modification to either the core or enhancement systems. In addition, the second stage can use an entirely different coding paradigm than the core stage, as is the case in this paper. The disadvantage is that the second stage can not access individual parameters specified by the bitstream r_c or used in the core encoder. Another disadvantage is that the core encoder is not necessarily tuned for use in a two stage structure.

Two Algebraic-Code-Excited Linear Predictive (ACELP) coders have been investigated for use as a core: G.729 at 8 kb/s, and G.723.1 at 5.3 kb/s. Both encoders have a pre-processing step which consists of a high pass filter and pre-scaling. Both decoders have a postfilter which enhances the quality of the synthesis output. Though the original goal is to make the second stage as independent of the core stage as possible, one concession has been made: the signal x represents the input after pre-processing, and s_c the core output before post-processing.

3. INITIAL INVESTIGATION

An initial step in the design of the second stage is the characterization of various signals of interest: x, the input signal; s_c, the

core decoder output; $\mathbf{e} = \mathbf{x} - \mathbf{s}_c$, the difference signal. The results of the investigation provide the key motivations for using a hybrid structure. All test files used are 16 bit 8 kHz samples. Only active segments in time for which the average signal energy of $\mathbf{x}$ exceeded a given threshold are used in the analyses.

The first measurements are LPC predictive gains. The linear predictive filters are defined every 10 ms on a 20 ms window of data using the autocorrelation method [7]. A 20 ms Hamming window is used. Signals $\mathbf{x}$, $\mathbf{s}_c$, and $\mathbf{e}$, can be used to define filters. Predictive gains are defined by applying the filters to various signals, using only the corresponding middle 10 ms of the 20 ms interval of time used in definition of the filter. The predictive gain $\mathcal{G}(\mathbf{H}, \mathbf{y})$ (in dB) for a filter $\mathbf{H}$ applied to a segment of data $\mathbf{y}$ is defined by: $\mathcal{G}(\mathbf{H}, \mathbf{y}) = 10log_{10}(\|\mathbf{y}\|^2 / \|\mathbf{H}.\mathbf{y}\|^2)$.

Average gains for a few signals are given in Table 1. G.723.1 is used to generate $\mathbf{s}_c$.

Table 1: Mean (segmental) Predictive Gains (dB)

Combination	1	2	3	4
Filter defined on:	$\mathbf{x}$	$\mathbf{e}$	$\mathbf{x}$	$\mathbf{s}_c$
Gain defined using:	$\mathbf{x}$	$\mathbf{e}$	$\mathbf{e}$	$\mathbf{e}$
Order	10	10	10	10
Clean female speech, 28 sec	8.98	4.65	2.27	0.46
Clean male speech, 28 sec	9.00	4.57	2.68	0.80
Synthetic Music, 17 sec	4.90	3.11	2.16	0.97
Brass and strings, 14 sec	7.61	4.42	2.92	1.74
Vocal music, 16 sec	4.32	2.32	-1.44	-0.38
Car noise, 8 sec	3.77	2.71	2.05	1.26

Combinations 1, 2 and 3 demonstrate that 10th order all pole modeling of the input and error spectra provide only marginal predictive gains for some signals, in particular with non-speech signals. Combination 4 is included as it uses a filter available to the second stage decoder without the need to use bits from r_e. The gains demonstrate that this filter is of minimal use for encoding $\mathbf{e}$. The results using higher LPC orders are similar.

Other investigations focused on signal to noise ratios between $\mathbf{x}$ and $\mathbf{e}$. Subband SNR's were computed using FFT's and MDCT's. Generally, the mean SNR's for many music files are lower than 4.5 dB in most frequency ranges, with components below 500Hz and above 2500Hz having the lowest SNR's. Often SNR's in the low frequency range change by as much as ± 15 dB on a 10 ms basis. These distortions appear to be the most important to reduce in order to improve the quality of the core for music.

For clean speech files, SNR's below 1.5 kHz are generally high, with means in the range of 8-10 dB. Above 2 kHz SNR's have means on the order of 3 dB. The swirling noise observed when coding input signals with car noise is probably the most difficult distortion to characterize. It has been conjectured that this phenomenon is related to random movements of the poles of the LPC filter [8]. The frequency range of 500 Hz to 2 kHz appears to be most significant to this phenomenon.

These and other analyses show that while the LPC model can provide significant coding gains for speech-like signals, it is not necessarily applicable to either more general signals like music, or the error signal from the core stage. In addition, subtle distortions like the swirling in car noise and those with music require more careful correction. The gross modeling of the LPC filter is not useful for this purpose. These observations, and the desire to use more advanced perceptual techniques to compensate for distortions introduced by the core, are the primary motivations for using a transform paradigm for the second stage.

4. CHOICE OF TRANSFORM AND DELAY

A Modified Discrete Cosine (Lapped) Transform (MDCT) [9], operating on 160 samples with a 80 (10 ms) sample overlap is used for the second stage. This enables synchronization with both G.729, G.723.1, and other ITU-T standards which use the 10 ms multiple for framing. G.723.1 has a 30 ms frame with a 7.5 ms lookahead. G.729 has a 10 ms frame with a 5 ms lookahead.

The maximum combined bitrate of the core and enhancement stages is set at 16 kb/s. This translates into 1 bit/input-sample for the enhancement layer with a G.729 core. For each frame of data coded by the second stage, overhead information consisting of gains and bit assignments need to be transmitted. Experiments suggest that this information requires about 10-20 bits/frame which is significant for a 10 ms frame.

To lower the overhead, two consecutive MDCT's are coded together and share common overhead information. This increases the frame size to 20 ms, but maintains a 10 ms overlap between frames. The resultant delay of the two stage coder using this paired scheme is given in Table 2. Delays using other unpaired schemes are also given.

Table 2: Two Stage Algorithmic Delays

MDCT overlap	**Delay with G.729**	**Delay with G.723.1**	**δ Delay over core**
10 ms unpaired	25 ms	47.5 ms	10 ms
20 ms unpaired	45 ms	67.5 ms	30 ms
10 ms paired	35 ms	57.5 ms	20 ms

5. THE SECOND STAGE CODER

The second stage coder enhances the core output by modifying the MDCT of the core stage output and performing an inverse MDCT. The bits r_e define the modification. There are 3 different time scales of framing: sub-subframes, subframes, and frames. The basic set of encoding and decoding operations follow this framing hierarchy. Sub-subframes occur every 5 ms and consist of the most recent 20 ms of data. Subframes occur every 10 ms, and consist of the most recent 20 ms of data. Frames consist of 2 consecutive subframes.

5.1. Second Stage Encoder

Only one operation is performed on a sub-subframe basis, a spectral estimate of the input signal computed via a FFT on the present 20 ms block of the input $\mathbf{x}$. The next operations are done on a subframe basis. Two MDCT's are calculated every subframe: $\mathbf{X}_k$, the MDCT of the present subframe k of the input $\mathbf{x}$; $\mathbf{S}_k$, the MDCT of the present subframe k of the core output $\mathbf{s_c}$. The present and past (sub-subframe) spectral estimates are used to define an acceptable noise threshold, $\mathbf{T}_k$, for subframe k. The calculation is based on principles outlined in [10].

Frames consist of 2 consecutive subframes, subframe $k-1$ and subframe k, and so occur every 20 ms. The frame operations begin with a pre-scaling of the coefficients of $[\mathbf{S}_{k-1}, \mathbf{S}_k]$ producing an new MDCT pair $[\tilde{\mathbf{S}}_{k-1}, \tilde{\mathbf{S}}_k]$. Low, mid, and high frequency bands, are scaled in different ways. The process is highly adaptive depending on how much pre-scaling is required. Musical sequences generally require more scaling than speech sequences. The coder allows for various degrees of pre-scaling, and produces a variable length bit stream from this process.

Following the scaling, the error MDCT, $[\mathbf{E}_{k-1}, \mathbf{E}_k] = [\mathbf{X}_{k-1}, \mathbf{X}_k] - [\tilde{\mathbf{S}}_{k-1}, \tilde{\mathbf{S}}_k]$, is calculated. A gain normalization envelope $[\mathbf{G}_{k-1}, \mathbf{G}_k]$ is selected. This envelope is defined using a smoothed version of $[\tilde{\mathbf{S}}_{k-1}, \tilde{\mathbf{S}}_k]$ combined with vector quantized gains in a

number of subbands. Four possible smoothing processes exist. The number of subbands is selected in a closed loop fashion. Having a choice allows the encoder to better balance the number of bits used for gain normalization to those used for other processes. The normalized MDCT, $[\mathbf{N}_{k-1}, \mathbf{N}_k]$, $\mathbf{N}_k(i) = \mathbf{E}_k(i)/\mathbf{G}_k(i)$, is then coded by a two stage vector quantization process. Four dimensional inter/intra-subframe VQ's are used.

The quantization process begins by assigning a fraction of the bits remaining from the total bit budget (after the pre-scaling and gain normalization) to the first stage vector quantizer (VQ). The remaining bits are assigned to the second stage vector quantizer. There are two possible splits in the bit assignment between the two quantizers. The actual split used is selected by a closed loop procedure.

One of two possible first stage VQ codebooks may be used: a 5 bit or a 7 bit codebook. The lower rate codebook allows more MDCT coefficients to be quantized, and the higher rate gives more accurate quantization. The choice used is signaled by a 1 bit flag.

Given the bit budget and codebook used, the bit assignment for the first stage VQ is implicit on $[\mathbf{G}_{k-1}, \mathbf{G}_k]$. Twelve extra bits of side information may be taken from the first stage VQ bit budget to modify this assignment. A one bit flag signals whether or not this is done.

First stage quantization followed by inverse quantization produces an approximation $[\tilde{\mathbf{N}}_{k-1}, \tilde{\mathbf{N}}_k]$ to $[\mathbf{N}_{k-1}, \mathbf{N}_k]$. The approximation $[\tilde{\mathbf{N}}_{k-1}, \tilde{\mathbf{N}}_k]$ and $[\mathbf{G}_{k-1}, \mathbf{G}_k]$ are used to assign the remaining bits for the second stage quantization. The second stage codebooks are tree structured relative to the first stage codewords, and so the second stage quantization procedure needs to know the first stage VQ indices. Second stage inverse quantization provides a final approximation $[\hat{\mathbf{N}}_{k-1}, \hat{\mathbf{N}}_k]$. This is inverse normalized and added to $[\tilde{\mathbf{S}}_{k-1}, \tilde{\mathbf{S}}_k]$ to provide a new approximation $[\hat{\mathbf{X}}_{k-1}, \hat{\mathbf{X}}_k]$. to $[\mathbf{X}_{k-1}, \mathbf{X}_k]$.

$$\hat{\mathbf{X}}_k(i) = \hat{\mathbf{N}}_k(i)\mathbf{G}_k(i) + \tilde{\mathbf{S}}_k(i). \quad (1)$$

Closed loop decisions are made by comparing $[\hat{\mathbf{X}}_{k-1}, \hat{\mathbf{X}}_k]$. to $[\mathbf{X}_{k-1}, \mathbf{X}_k]$ for different codebook, bit assignment, and gain normalization, options. During this process it is often the case that similar vector quantization indices are shared by different combinations of options. This fact is used to simplify the closed loop procedure. The final encoding structure is illustrated in Figure 2.

Table 3: Bit-Stream Format of Second Stage Encoder

Operation	Comments		Bits
Standard Scaling	Low Freq		6
i_n	All Freq		3
Selective Scaling	Low Freq	G.723.1	2-41
i_p		G.729	2-29
	Mid Freq		1 - 9
	High Freq		1 - 10
Closed Loop (CL)	Type of Gain Normal.		1
Selections	Type of Stage 1 Bit Assgn.		1
i_c	Stage 1 VQ codebook		1
	Stage 2 VQ bit budget B		1
Gain Norm. i_g	Selected in CL		4 or 16
Smoothing Process	For raw gain envelope.		2
Stage 1 Bit Assgn. i_o	Selected in CL		0 or 12
2 stage VQ	B selected by CL		
i_f	1^{st} stage VQ indices		left-B
i_s	2^{nd} stage VQ indices		B

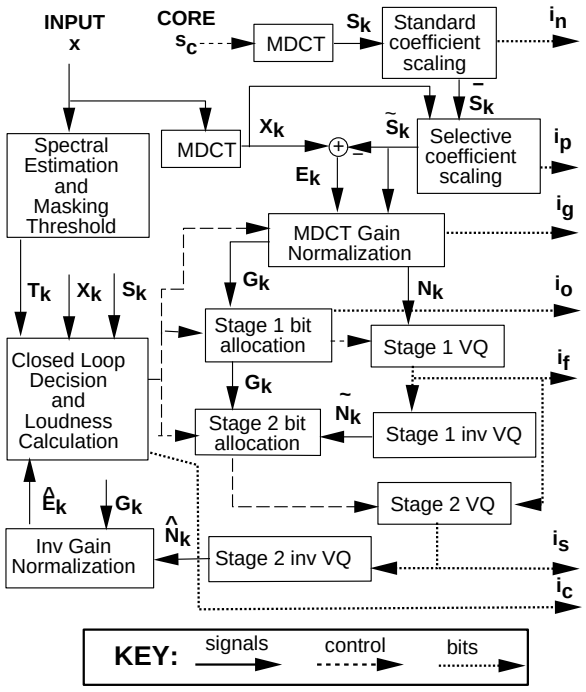

Figure 2: Second Stage Encoder

5.2. Second Stage Bit Stream

The final bitstream format of the encoder is adaptive. The adaptation is done in both closed and open loop fashions. The (variable) format is shown in Table 3. Bits within the Selective Scaling and Closed Loop Selection streams define the adaptation. The main differences between the G.729 and G.723.1 formats are the number of bits used for the low frequency scaling.

5.3. Second Stage Decoder

The decoding process begins by calculating $[\mathbf{S}_{k-1}, \mathbf{S}_k]$. Scaling is done to produce $[\tilde{\mathbf{S}}_{k-1}, \tilde{\mathbf{S}}_k]$. The gain normalization envelope $[\mathbf{G}_{k-1}, \mathbf{G}_k]$ is calculated, and first stage bit assignment is recovered using this envelope and possibly 12 extra correction bits.

The error approximation $[\tilde{\mathbf{N}}_{k-1}, \tilde{\mathbf{N}}_k]$ is recovered by the inverse first stage vector quantizer. This approximation and $[\mathbf{G}_{k-1}, \mathbf{G}_k]$ define the bit assignment for the second stage vector quantizer. The final approximation $[\hat{\mathbf{N}}_{k-1}, \hat{\mathbf{N}}_k]$ is recovered by inverse second stage vector quantization. This is inverse normalized to give the error estimate.

The error estimate is added to the scaled MDCT $[\tilde{\mathbf{S}}_{k-1}, \tilde{\mathbf{S}}_k]$ resulting in $[\hat{\mathbf{X}}_{k-1}, \hat{\mathbf{X}}_k]$. The inverse MDCT (IMDCT) of $[\hat{\mathbf{X}}_{k-1}, \hat{\mathbf{X}}_k]$ is put through the add and overlap process, and the result is post-processed by a high pass filter. The final output is $\mathbf{s}_e$.

5.4. Self Embedded Property

Another property of the enhancement stage is that its bitstream is also embedded. At the Standard and Selective Scaling stages of the decoding process there are either incomplete or complete versions of the signals $[\bar{\mathbf{S}}_{k-1}, \bar{\mathbf{S}}_k]$ and $[\hat{\mathbf{S}}_{k-1}, \hat{\mathbf{S}}_k]$. If the final bitstream reaching the decoder ends in/at these stages, the incomplete/complete version of $[\bar{\mathbf{S}}_{k-1}, \bar{\mathbf{S}}_k]$ or $[\hat{\mathbf{S}}_{k-1}, \hat{\mathbf{S}}_k]$ can be sent to the IMDCT (with $[\hat{\mathbf{N}}_{k-1}, \hat{\mathbf{N}}_k]$ assumed zero) to produce an enhanced output.

Bits for the Closed Loop Choice, Gain Normalization, and First Stage Bit Assignment, by themselves can not be used to improve on the reconstruction. However, the inverse VQ processes

can be stopped at any point in the calculation of either $[\tilde{\mathbf{N}}_{k-1}, \tilde{\mathbf{N}}_k]$ or $[\hat{\mathbf{N}}_{k-1}, \hat{\mathbf{N}}_k]$. Partial versions of either of these vectors can be inverse normalized and combined with $[\tilde{\mathbf{S}}_{k-1}, \tilde{\mathbf{S}}_k]$ to produce a reconstruction via the IMDCT.

In cases where processing power is a limitation, the embedded property also allows both the second stage encoder and decoder to produce or use only partial portions of the second stage bitstream. In this way, the structure is also scalable in complexity.

6. A-B COMPARISON TESTS

To compare the performance of this system to other fixed rate standard coders informal A-B comparison tests were conducted. Two standard coders were used for the comparison: G.728 at 16 kb/s, and the GSM Enhanced Full Rate (EFR) coder at 12.2 kb/s. The total rate of the two stage structure is fixed at 16 kb/s. The tests used 10 files: 6 clean speech, 3 male, and 3 female; 5 music files, synthesized music, orchestral music, a saxophone solo, and 2 vocals pieces with background music; 2 mixed inputs, one female speech with classical music, the other female speech with car noise at -15dB relative to the speaker. The core coder used is G.723.1. Nine listeners were used.

The fraction of trials in which the two stage embedded structure is preferred over each of the standard coders is shown in Table 4.

Table 4: Preference of 2 Hybrid Structure over Standard Coders

Fraction of Trials the Hybrid Structure is Preferred			
Input		**G.728**	**GSM-EFR**
Speech:	Female Speech	18/27	14/27
	Male Speech	11/27	13/27
Music:	Synthesized	2/9	8/9
	Orchestra	3/9	7/9
	Saxophone	1/9	6/9
	Vocal piece 1	6/9	7/9
	Vocal piece 2	3/9	7/9
Mixed:	Female+classical	4/9	4/9
	Female+car noise	1/9	2/9

The tests show, as expected, that the performance of the two stage structure depends strongly on the performance of the core stage. For clean speech the core speech coder is making a good use of the core bitrate. In this case the two stage structure has a performance practically equal to that of G.728. However, for general inputs such as music, the structure at 16 kb/s has a performance between the 12.2 kb/s GSM-EFR coder and the 16 kb/s G.728 coder. Informal tests show that the performance of the hybrid structure with a G.729 core is less than that using the G.723.1 core. The primary reason for this is that the improved performance of the G.729 core over the G.723.1 core does not make up for 8.0-5.3=2.7 kb/s loss in bitrate available to the second stage.

7. CLOSING REMARKS

The general two stage structure provides a means of designing embedded coding structures, and a way to enhance an existing speech/audio coding system. The hybrid nature of the structure allows the structure to compensate for distortions inherent to the core coding paradigm, while still taking advantage of the high compression ratios the core provides on certain inputs like speech. The structure is also flexible and can adjust to various levels of complexity, bitrate, and coding quality, both during and after encoding. The present implementation allows for two possible algorithmic delays.

The second stage coder is highly adaptive. Though this enables the coder to adapt to various types of distortions and make efficient use of bits, it also makes the second stage encoder very sensitive to bit errors. The second stage should only be used in cases of very low bit error rates. It is also worth mentioning that two stage structures are also sensitive to bit errors in the core stage. For example, a sign change in the output of the core stage will not affect the core output quality, but can make a large difference on the combined output of the two stages.

The largest room for improvement in the 2 stage structure is in the core coder. The core coders used in this study often produce distortions that are extremely difficult to compensate for, requiring an excessive expenditure of bits by the second stage. Modification of the core was not considered in the present work.

ACKNOWLEDGMENTS

I would like to acknowledge my colleagues at Bell Laboratories, in particular Dr. Peter Kroon, for their support and advice while doing this work. I would also like to acknowledge similar concurrent work being done in MPEG-4 standardization process.

8. REFERENCES

[1] ITU-T. *Dual Rate Speech Coder for Multimedia Communications Transmitting at 5.3 and 6.3 kbit/s*, March 1996. Recommendation G.723.1.

[2] ITU-T. *Coding of Speech at 8 kbit/s Using Conjugate-Structure Algebraic-Code-Excited Linear Prediction (CS-ACELP)*, March 1996. Recommendation G.729.

[3] A. Le Guyader and E. Boursicaut. Embedded wideband VSELP speech coding with optimized codebooks. In *IEEE Workshop on Speech Coding for Telecommunications*, pages 15–16, Quebec, Canada, October 1993.

[4] R. D. De Iacovo and D. Sereno. Embedded CELP coding for variable bit-rate between 6.4 and 9.6 kbit/s. In *IEEE Int. Conf. of Acoustics, Speech, Signal Processing*, pages 681–684, Toronto, May 1991.

[5] S. Zhang and F. Lockhart. Embedded RPE based on multistage coding. *IEEE Trans. on Speech and Audio Proc.*, 5(4):367–371, July 1997.

[6] B. Tang, A. Shen, A. Alwan, and G. Pottie. A perceptually based embedded subband speech coder. *IEEE Trans. on Speech and Audio Proc.*, 5(2):131–140, March 1997.

[7] L. Rabinar and B-H. Juang. *Fundamentals of Speech Recognition*. Prentice Hall, Englewood Cliffs, New Jersey, 1993.

[8] T. Wigren, A. Bergstrom, S. Harrysson, F. Jansson, and H. Nilsson. Improvements of background sound coding in linear predictive speech coders. In *IEEE Int. Conf. of Acoustics, Speech, Signal Processing*, pages 25–28, 1995.

[9] J. P. Princen and A. B. Bradlen. Analysis and synthesis filter bank design based on time domain alaising cancellation. *IEEE Trans. on Acoustics, Speech, and Signal Proc.*, 34(5):277–284, May 1986.

[10] J. D. Johnston. Transform coding of audio signals using perceptual noise criteria. *IEEE Journal on Selected Areas in Communications*, 6(2):314–323, Feb 1988.