國立交通大學

電子工程學系

碩士論文

一個雜訊遮罩比最佳化的音訊位元率轉碼技術

NMR Optimized Bitrate Transcoding for

MPEG-2/4 AAC with LC Profile

研 究 生：賴德亘

指導教授： 蔣迪豪 博士

中 華 民 國 九 十 五 年 六 月

一個雜訊遮罩比最佳化的音訊位元率轉碼技術

NMR Optimized Bitrate Transcoding for MPEG-2/4 AAC with LC Profile

研究生：賴德亘　　Student: Te-Hsueh Lai

指導教授：蔣迪豪　　Advisor: Tihao Chiang

國 立 交 通 大 學

電 子 工 程 學 系　　電 子 研 究 所 碩 士 班

碩 士 論 文

A Thesis

Submitted to Department of Electronics Engineering & Institute of Electronics

National Chiao Tung University

in partial Fulfillment of the Requirements

for the Degree of

Master

in

Electronics Engineering

June 2006

HsinChu, Taiwan, Republic of China

中華民國 九十五 年 六 月

一個雜訊遮罩比最佳化的音訊位元率轉碼技術

研究生: 賴德亘　　指導教授: 蔣迪豪 博士

國立交通大學
電子工程學系 電子研究所碩士班

摘要

多媒體應用例如音樂隨選、數位廣播等愈來愈普及。為了在異質性網路中傳送多媒體內容到多樣性客戶端，即時音訊串流技術被廣泛地應用。為了適應不同的網路狀況與客戶端的設備和功能，音訊串流技術採用位元率的轉換技術。本論文提出一個可以應用於 MPEG-2/4 AAC-LC 的標準位元流之快速且雜訊遮罩比最佳化的位元率轉碼技術(Fast Noise-to-Masking Ratio optimized bitrate transcoding， FRDOT)。對於每一個設定的位元率，FRDOT 找出每個頻帶最佳的量化參數，以達成每個頻帶內的雜訊遮罩比(Noise-to-Masking Ratio, NMR)之最佳化。並且，基於音訊位元流在位元轉碼前後具有相同聽覺遮罩 (Masking thresholds)的原理，轉碼技術最佳化的準則可以從雜訊遮罩比推演到雜訊訊號比(Noise-to-Signal Ratio, NSR)。基於雜訊訊號比，為了加速最佳化的位元率轉碼技術，我們採用表格查詢的方法以減少總運算量。為了進一步加速轉碼器，FRDOT 採用頻寬限制器以減去在編碼區塊中最佳量化參數的遞迴式搜尋法所需的時間。並且，FRDOT 提出位元流控制模組，使得轉碼器的輸出位元率更接近目標位元

率。實驗結果顯示，本論文所提出的位元率轉碼器可將音訊位元流從高位元率轉換至較低位元率，並且相較於串接轉碼器有半分貝到三分貝的雜訊遮罩比改善。在執行時間方面，則有五到八倍的加速。

關鍵詞: 雜訊遮罩比，位元率適應，音訊轉碼器，聽覺遮罩，位元率失真量最佳化，MPEG AAC LC

NMR Optimized Bitrate Transcoding for MPEG-2/4 AAC with LC Profile

Student: Te-Hsueh Lai Advisor: Dr. Tihao Chiang

Department of Electronic Engineering &
Institute of Electronics
National Chiao Tung University

Abstract

Real-time audio streaming services like music-on-demand (MOD), digital audio broadcasting (DAB), etc, deliver multimedia content over heterogeneous networks and to client devices with varying capabilities. To fit the network conditions and the clients' capabilities, the bitrate adaptation based on the transcoding techniques is applied. We present a noise-to-masking-ratio (NMR) optimized MPEG-2/4 AAC LC transcoder, which is called as Fast Rate-Distortion Optimized Transcoder (FRDOT). In addition, FRDOT searches for the optimal scalefactor under the NMR criterion at a given bitrate. The computation of NMR difference is replaced by the derivation of signal-to-noise-ratio (SNR) difference since the audible masking thresholds of the input and output bitstreams are identical before and after transcoding. Within FRDOT transcoder, the SNR value is further converted to a noise-to-signal-ratio (NSR) to represent the distortion energy of audio signals. Therefore, the NMR optimized transcoding can be converted to the NSR optimized transcoding. The NSR optimized transcoding can find the optimal scalefactor increment according to the magnitudes of quantized input coefficients and the target bitrate. To speed up the search of optimal

scalefactor increment, a table lookup technique is used. To further reduce the execution time, the bandwidth limiter is adopted to remove the iterative rate-distortion optimization of a frame. In addition, a bitrate control module is proposed to make the averaged bitrate of output bitstream close to the target bitrate. The experiment results show that the NMR value of FRDOT is better than the NMR value of cascaded transcoder (CT) by 0.5-3.0 dB at different bitrates and FRDOT can speed up CT by 5-8 times on the average.

Key words: noise-to-masking ratio (NMR), bitrate adaptation, transcoder, masking threshold, rate-distortion optimization (RDO), MPEG AAC LC, fast transcoding

誌謝

因為有身旁許多人的指導與鼓勵，我才能順利完成這篇論文。首先我要謝謝我的指導教授蔣迪豪老師，讓我能在這麼堅強的學術團隊裡學習與成長。

另外，也很感謝杭學明老師，俊能、俊毅、政翰、繼大學長以及育彰(屎蛋)給予我音訊研究方面的指導與建議。尤其俊能學長與屎蛋。俊能學長總是在百忙之中，仍全心全意地與我討論研究內容，給予我寶貴的意見，我由衷地感謝能在俊能學長的幫助下完成這篇論文。接著要謝謝屎蛋同學，與你的討論與知識交流，是我研究音訊的最佳動力。

也謝謝 Commlab 實驗室的同學、身邊好友們的加油與鼓勵，能夠認識大家，我的研究生生活才能有這麼多采多姿的回憶。最後感謝我的父母、家人給予我的關心與照顧，有了你們，我會是最幸福的長男。

「希望我的朋友與家人都能幸福與健康！」

2006 年 7 月 14 日，內蒙輝騰希勒大草原熬包上的願望。

德亘

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In this chapter, we introduce the demand of content adaptation for multimedia delivery applications. To realize the content adaptation, motivation of using transcoding techniques is described. In addition, the key issues to realize the transcoders to fit some real-time application scenarios are introduced.

The multimedia delivery over networks becomes more and more popular recently. Media providers offer real-time audio streaming services covering music-on-demand (MOD), digital audio broadcasting (DAB), etc. As shown in Figure 1-1, Mobility and Service Adaptation in Heterogeneous Mobile Networks (MASA) QoS Framework is a joint project of NEC Europe Ltd, Heidelberg, Siemens AG Munich and the University of Ulm. MASA framework includes a distributed set of autonomous QoS Brokers. Each Broker is responsible for the brokerage between managers with different tasks, which present resources, network media, monitoring, policy management and mobility management. In a mobile environment with heterogeneous devices that are connected via heterogonous networks, media adaptation can be viewed as a comprehensive quality-of-service (QoS) mechanism.

In addition, for streaming high quality audio, the required bitrates are typically 64 to 128 kilo-bits per second (kbps). To deliver audio bitstreams of high bitrates over long range (from continent to continent) or over wireless network [3] is still a challenge. The key issue is to reliably provide high quality audio services over varying bandwidth and heterogeneous networks including GSM, Wireless LAN, UMTS, etc. For example, for wireless communication services, the client's spatial locations (indoor or outdoor) and the client's velocities (stationary or in motor vehicles) both affect the transmission capacity [1]. Even in a single connected network session, the bandwidth fluctuates during the transmission session. In case of network congestion, the packets may be lost or delayed, which is not acceptable at real-time applications. To prevent packet loss as the channel bandwidth decreases or the network is congested, the audio services should seamlessly lowers the bitrates of delivering bitstreams. The dynamical adaptation of bitstream bitrates is required to simultaneously serve multiple clients with different capacities of receiving, decoding and playback as in MOD

applications.

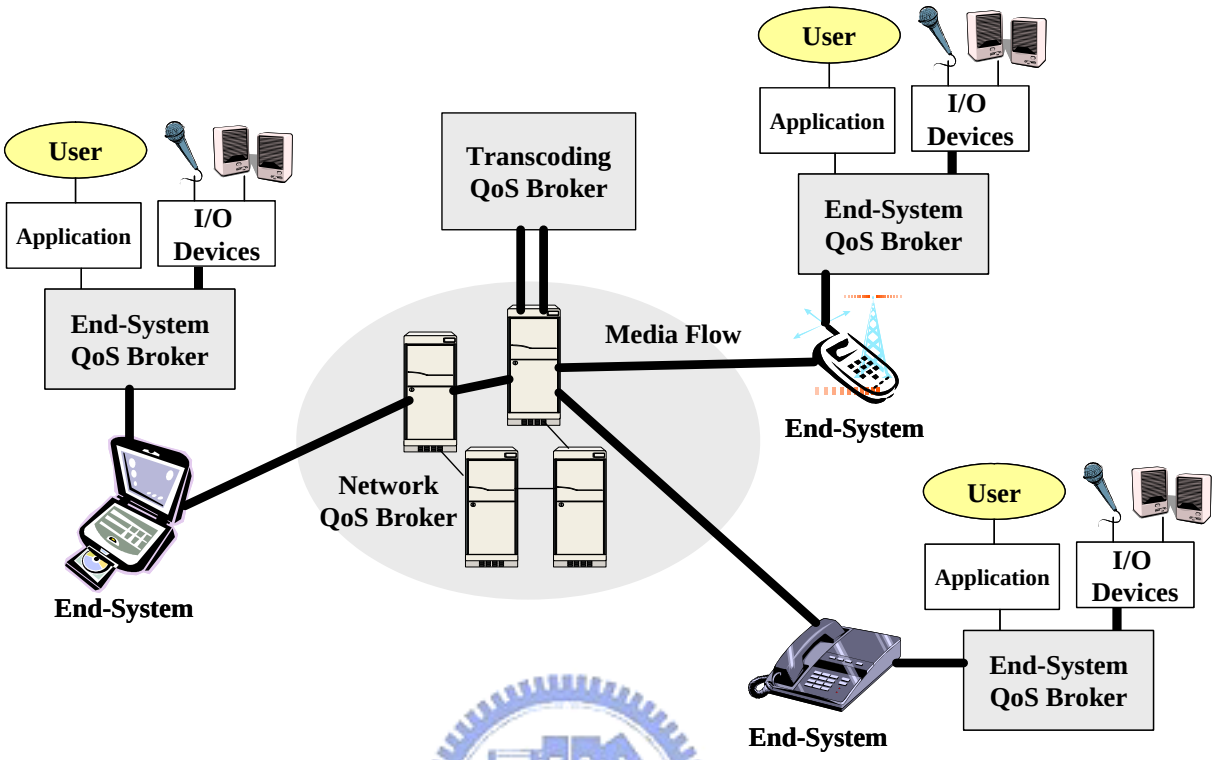

Figure 1-1. Architecture of Mobility and Service Application (MASA) [9]

Bitrate adaptation approaches [8]-[10] have been adopted to trade audio quality for transmission throughput in real-time content delivery applications. The bitrates of audio bitstreams to deliver are adapted according to the channel bandwidth and receivers' capacities. In addition, to save the storage space to store a single audio sequence in bitstream formats of distinct bitrates, a unitary bitstream of the specified format is archived at the server database.

For audio streaming, the archived bitstreams can be generated by scalable coding and non-scalable coding. For scalable audio coding, bit slice arithmetic coding (BSAC) defined at MPEG-4 specification of version 2 could be used. BSAC replaces the original AAC Huffman coding by the arithmetic coding. With the side information for scalability, the coding performance of BSAC is generally quite a bit lower than non-scalable AAC coding at the same rate. Thus from the rate-distortion performance viewpoint, non-scalable AAC coding scheme is adopted and the bitrate adaptation is done by transcoding techniques that directly convert the compressed bitstream from high bitrates to low bitrates [18]-[20]. As shown in Figure 1-1, Transcoding QoS Brokers allow rate adaptation schemes at the server in the end

systems for heterogeneous clients or special network links. In an end-to-end audio streaming scenario, the mechanism of Transcoding QoS Broker can be realized by porting the audio transcoding techniques onto the MASA architecture.

Existing applications that use the bitrate transcoding techniques for audio streaming consist of Shoutcast [11] and AllofMp3 [12]. Shoutcast [11] constructed internet radio stations where the audio signals are usually compressed at high bitrates prior to low bitrates transmission. AllofMp3 [12] that started with a 384 kbps MP3 archive and recently updated to lossless source files presents an online music store. In addition, AllofMp3 also supports the "Preview" and "Online encoding" functionalities. "Preview" enables consumers to listen to the music before buying. Consumers can download a low quality sample or listen to the selected audio sequences immediately in a streaming audio format. "Online encoding" offers download options of bitstream formats and varying qualities at specified bitrates. The formats cover MP3, WMA, OGG, AAC, etc. The audio sequences are compressed as MP3 bitstreams at 192 kbps by default. Thus, the bitrate transcoding techniques can offer audio streaming of optional quality levels over heterogeneous networks to meet the users' demands and payment. For a given bitrate, the reaming issue is to retain the maximal rate-distortion performance of the transcoded bitstreams with lowest computation power and least storage space.

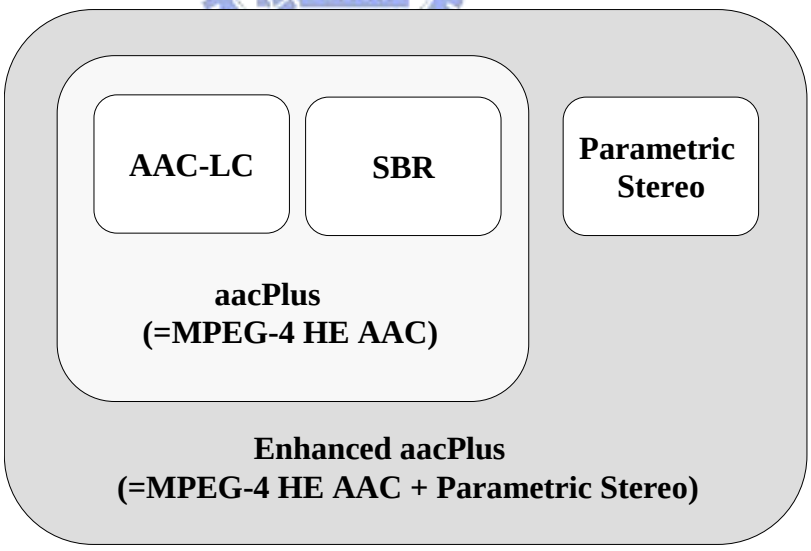

Figure 1-2. MPEG tool used to form the Enhanced aacPlus codec [5].

To evaluate the transcoding techniques on the audio content adaptation, we built a platform based on MPEG-2/4 Advanced Audio Coding (AAC) [7]. MPEG-2/4 Advanced

Audio Coding (AAC) [7] is one of the most recent audio coders specified by ISO/IEC MPEG standards committee. MPEG-2/4 AAC has been widely applied in the compression for PC-based and portable devices, compression for terrestrial digital audio broadcast, streaming of compressed media for both Internet and mobile telephone channels [4]. As shown in Figure 1-2, AAC is also the core of High Efficiency AAC (HE-AAC) that has been applied to both satellite-delivered digital audio broadcast and mobile telephony audio streaming [4].

To trade the rate-distortion performance for real-time transmission, we proposed a fast NMR Optimized transcoding algorithm in a novel transcoder that is called as Fast Rate-Distortion Optimized Transcoder (FRDOT). The performance was evaluated on a transcoding platform based on MPEG-2/4 AAC with LC profile. The results showed that FRDOT can retain good coding performance with less complexity as compared to cascaded transcoding (CT) that cascades a full decoder with a full encoder, which takes a great amount of computation to provide the best rate-distortion performance.

Chapter 2 reviews the major coding modules of MPEG-2/4 Advanced Audio Coding (AAC). Some homogeneous AAC-based transcoders that can support the bitrate scalability will be introduced. Chapter 3 describes the detailed algorithm of FRDOT. To optimize FRDOT, we modified Single Layer ACC Transcoding (SLAT) by refining the processing flows and presented a bitrate control module (BCM) for SLAT to meet with the target bits. To further improve the transcoding performance at low bitrates, a new algorithm to optimize the noise-to-masking ratio (NMR) is proposed. Chapter 4 gives experimental results using FRDOT. For advanced analyses, the experiment environment and the performance measures are introduced first. The performance comparison between FRDOT and the other transcoders is based on the rate control efficiency, the audio quality of converted bitstream and the execution time. Chapter 5 highlights the innovations of the fast NMR optimized transcoding algorithm and draws some future work on the possible application of FRDOT to advanced audio coding standards.

Chapter 2

MPEG-2/4 Advanced Audio Coding and Transcoding Techniques

This chapter reviews major coding modules of MPEG-2/4 Advanced Audio Coding (AAC). Some homogeneous AAC-based transcoders that can support the bitrate scalability will be introduced.

2.1 MPEG-2/4 AAC Major Coding Modules

In the developing history of Moving Picture Expert Group (MPEG) Audio, MPEG-1 audio coding specification has been standardized in 1992. It was the first world-wide standard for perceptually high quality audio coding. Recently, MPEG-1 layer III (MP3) audio format has become one of the most popular formats that can deliver music of Compact Disk (CD)-like audio quality on the internet. In 1994, for a higher quality audio standard, MPEG-2 Audio committee defined a new coding specification that is not backwards compatible with MPEG-1. The new standard, which is called MPEG-2 Advanced Audio Coder (AAC), was completed in April of 1997. MPEG-2 AAC tools are adopted as part of the kernel for MPEG-4 general audio coding. MPEG-4 general audio improves MPEG-2 AAC performance by adding perceptual noise substitution (PNS) and long term prediction (LTP) tools [1]-[2].

A typical AAC system consists of several separated components served as a series of self-contained tools. Based on the tradeoff between audio quality and system complexity, the AAC system offers three profiles covering Main Profile, Low Complexity (LC) Profile, and Scalable Sampling Rate (SSR) Profile. The Main Profile configuration provides the best quality and needs more memory and computational power than LC Profile and SSR Profile. The SSR Profile is used in case that scalable signal is required. In addition, the SSR Profile has lower complexity than Main Profile and LC Profile. Among the three profiles, the LC Profile configuration that can retain a high quality with considerably reduced memory and power requirement becomes the most widely used [13].

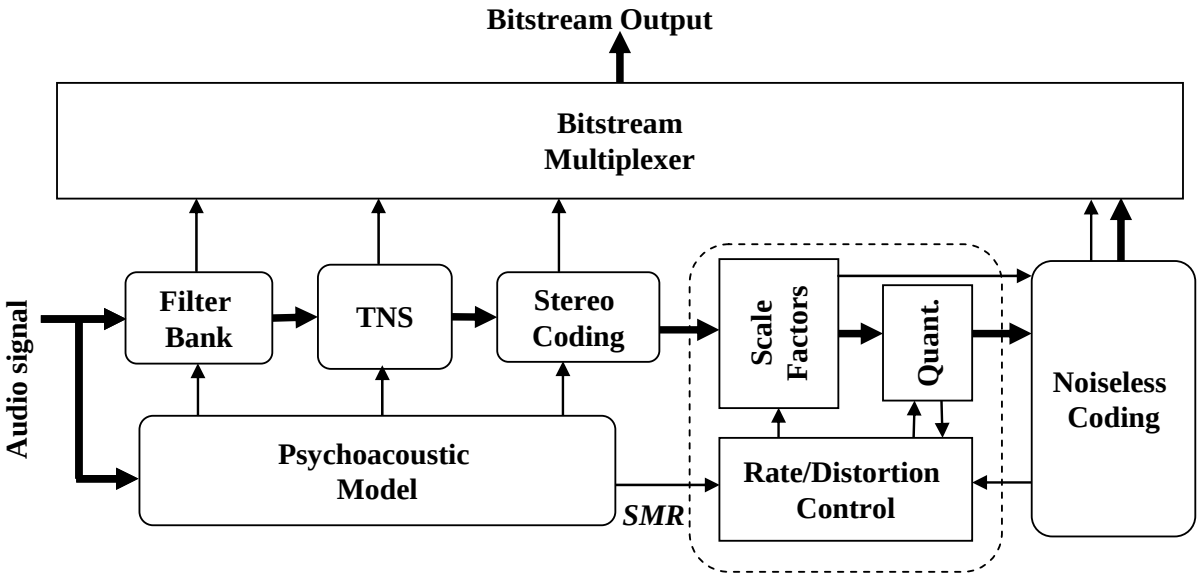

Figure 2-1. Block diagram of AAC-LC encoder

Figure 2-1 shows the block diagram of an AAC LC Profile encoder. The main modules include the Psychoacoustic Model, the Filter Bank, the Quantization, the Rate-Distortion Control, the Noiseless Coding, the Temporal Noise Shaping (TNS) and the Joint Stereo Coding. The details of each module are described at the following sections.

2.1.1 Psychoacoustic Model

A psychoacoustic model (PAM) is the essential module of perceptual codec [14]. The relationships between acoustic stimuli and human hearing are introduced into the control of perceptual distortion. The perceptual distortion is controlled by analyzing a psychoacoustic signal to estimate signal masking power. The masking thresholds of the psychoacoustic model present the just audible distortion at each point in the time-frequency plan. Considering the just audible distortion, the quantization distortion of the time-frequency parameters is not heard by human ears.

Some empirical characteristics of human ears that have been utilized in state-of-art audio coders will be briefly described in [14].

A. Hearing range and threshold

Human ears can hardly hear the sound with the frequency below 20 Hz and above 20 kilo-Hz. Even the sound frequency in the hearing range, human ears can not perceive sounds

below the “threshold in quite”. Therefore, the “threshold in quite” represents the lowest sound level that can be heard at a specific frequency as shown in Figure 2-2.

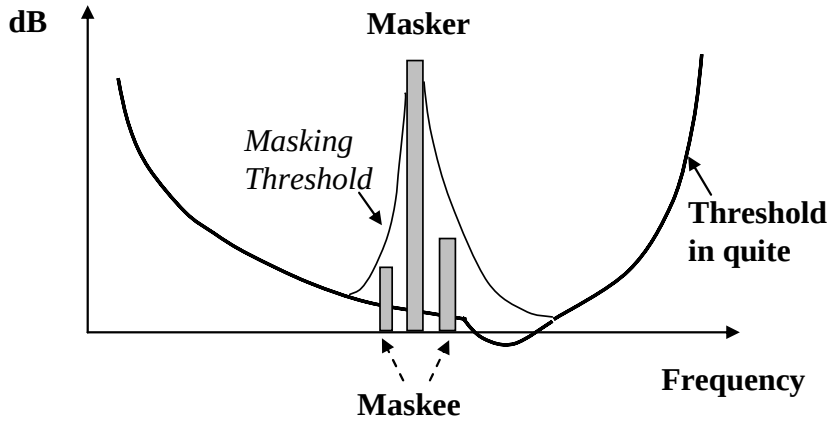

Figure 2-2. Example of frequency masking

B. Frequency masking

As illustrated in Figure 2-2, an audio signal below the masking threshold, which is defined as a Maskee, will not be audible when a Masker signal of close frequencies is presented. When the Masker signal is presented, a new post-masking threshold will be produced behind the Masker in the frequency domain. In addition, a pre-masking threshold with a much sharper slope is also produced in front of the Masker.

C. Temporal masking:

When a louder sound (Masker) occurs, a softer sound (Maskee) is simultaneously masked. Figure 2-3 shows that the Maskee 2 is masked as the Masker is presented. When the Masker is removed, the Maskee 3 is masked by a post-masking phenomenon. An unexpected pre-masking phenomenon that Maskee 1 was masked before Masker presents, is also important for psychoacoustic audio coding. In addition, the short period of pre-masking threshold makes “pre-echo” distortion to occur more easily.

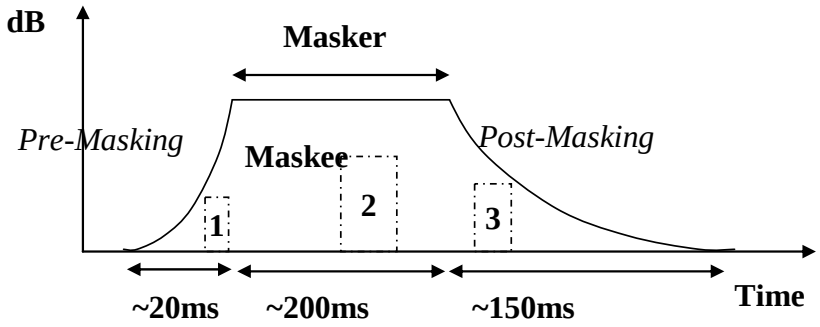

Figure 2-3. Example of temporal masking

D. Critical Bands

Human ears have different sensitivities to audio signals in different frequency bands. The sensitivities can be represented by about 20 frequency bands, which are named as "critical bands". The main concept of the critical band is that a masker exhibits a constant masking level in a critical band uncorrelated to the type of masker.

The characteristics of the human hearing system are utilized in perceptual audio coders. A certain level of coding distortion is not perceived by human hears as long as the noise signal is below the masking threshold estimated by the existing psychoacoustic models. The masked coding noise can be served as irrelevancy related to the human hearing. By removing the irrelevancy, the coding efficiency can be significantly improved.

MPEG-2/4 AAC employs a perceptual model similar to the "Psychoacoustic Model 2" in MP3 coder [6] in Figure 2-4. The "Psychoacoustic Model 2" analyses the input audio signals in the spectrum domain and gives an estimate of masking thresholds that determine the just noticeable level of quantization noise per frequency band. With the just noticeable noise levels, AAC coders can allocate different amount of bits to the varying frequency bands to increase rate-distortion performance.

The maximum distortion energy as the perceptual masking threshold is derived based on the three factors [6], which mean the shifting length for the threshold calculation, the window of psychoacoustic calculation and a constant sampling rate. The shifting length for the threshold calculation process is denoted as '*iblen*'. The values of '*iblen*' are 1024 and 128 for long Fast Fourier Transform (FFT).and short FFT respectively. For each FFT type, the newest *iblen* samples of the signal are delayed in the filter banks or the psychoacoustic calculation. The window of the psychoacoustic calculation is centered at the time window of the

time/frequency transform at the encoder. To reduce the overall encoding time, the psychoacoustic calculation is based on the specified set of tables that are built with the audio signals at the standard sampling rates. Thus, the audio signals with constant sampling rates are used in the calculation process.

There are four outputs from the psychoacoustic model. The first output is a set of Signal-to-Mask Ratios (SMR) and thresholds, which are used at the outer loop of the encoder to maximum the perceived quality. Subsequently, the modified discrete cosine transformation (MDCT) block type and the delayed time-domain data (PCM samples) used by the MDCT are used for the transformation. An estimate of the total bits per frame is input to the inner loop of the encoder to derive the quantization step-size.

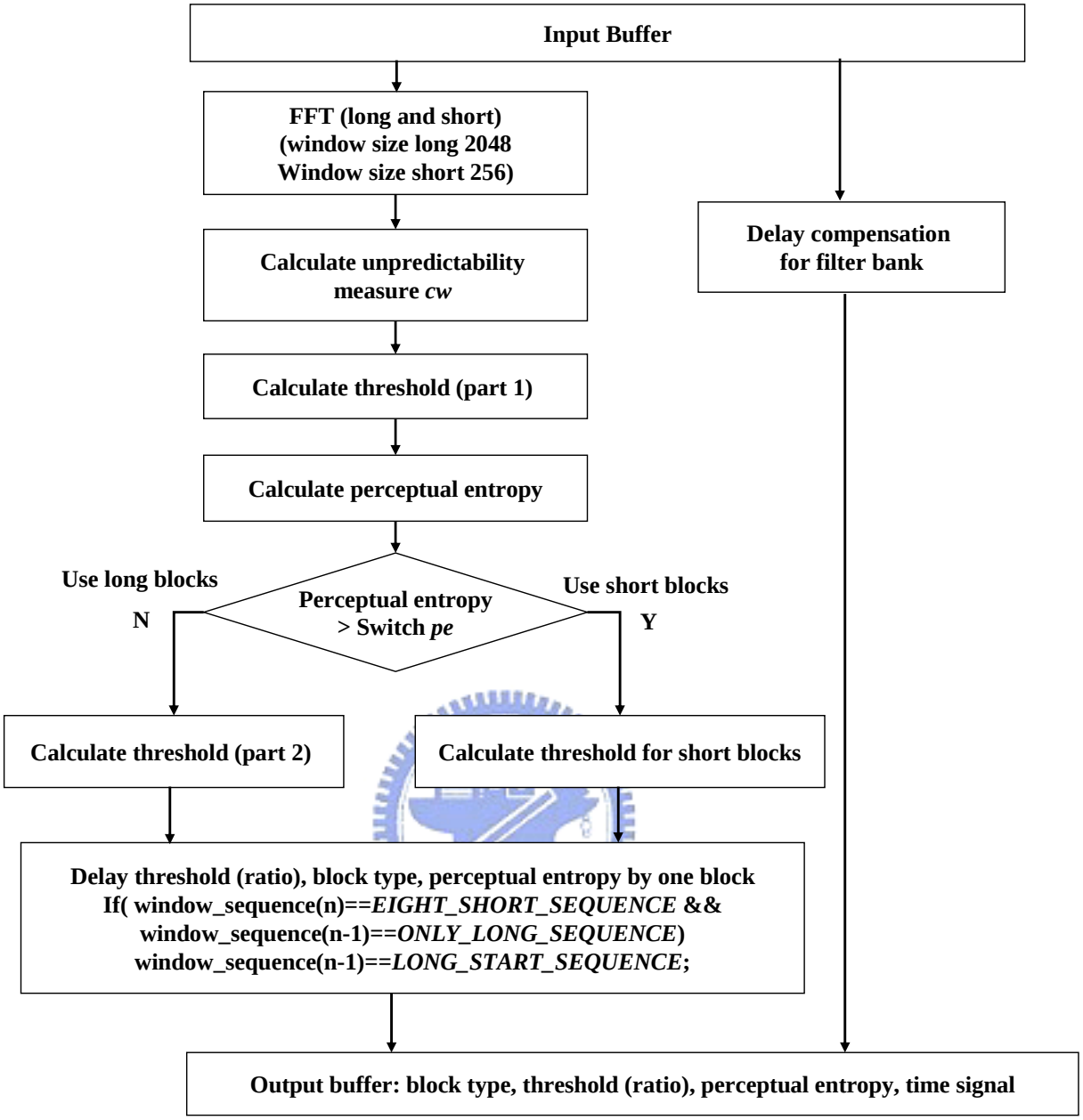

Figure 2-4. Block diagram of Psychoacoustic model 2 [6]

Psychoacoustic Model 2 transforms the spectral data into a “partition” domain. At each partition, the fractions of the tonal and non-tonal components are estimated to determine the levels of masking thresholds.

2.1.2 Filter Bank

The filter bank transforms input signals at the time domain into the internal signals at the spectrum domain. The spectral coefficients are used for coding. For adaptive audio encoding,

every filter bank covers different number of samples, modulates the samples by an adaptive window shape, and performs time to frequency domain mapping.

With 2048 time-domain samples, the coding efficiency is high. The coding performance is decreased by a phenomena called as pre-echo in compressing transient signals. AAC system fixed the pre-echo problem by choosing the length of transformation block based on the characteristics of input audio signals. The block length switching scheme adaptively takes 2048 samples for long window transform and 256 samples for short window transform. In addition, AAC encoders have 1024 and 128 spectral coefficients respectively for different window sizes.

To deriver the block type information from PAM, AAC takes two types of transition windows, *LONG_START_SEQUENCE* and *LONG_STOP_SEQUENCE*, to fix the long block and short block misalignment between audio coding frames.

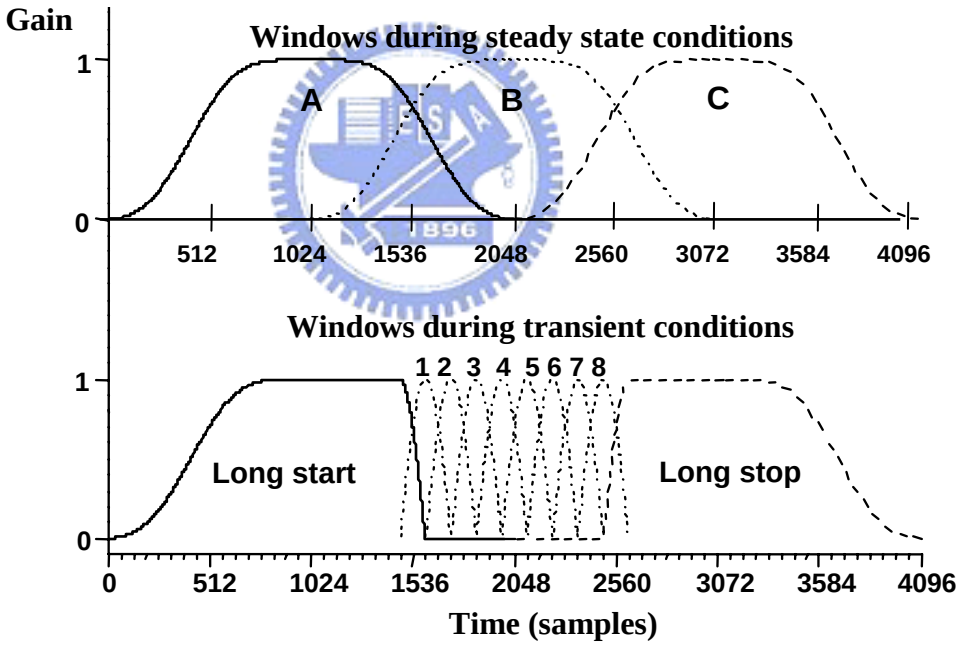

Figure 2-5. AAC block switching process [6].

For coding the blocks of varying sizes, modified discrete cosine transformation (MDCT) is applied based on a time-domain aliasing cancellation (TDAC) technique. The modulating window function affects the resolution of MDCT filter banks. AAC filter bank allows a change in the window shape to fit the input signal conditions. For stationary and transient

audio signals, AAC specification adopts two windows including sine window with a wider main lobe and Kaiser-Bessel-derived window (KBD) with improved far-off rejection respectively. In addition, each block of input samples is overlapped by 50% with the immediately preceding block and the subsequent block in order to reduce the boundary effect. To avoid the increase of system data rate by the overlapping, the reconstruction of audio samples is done by MDCT.

$$X_{i,k} = 2 \cdot \sum_{n=0}^{N-1} z_{i,n} \cos\left(\frac{2\pi}{N} \left(n + n_0 \right) \left(k + \frac{1}{2} \right) \right) \quad for\ 0 \le k < \frac{N}{2}, \tag{1}$$

where $z_{i,n}$ is the windowed input sequence, n is the sample index, k is the spectral coefficient index, i is the block index, N is the transformation window length and n_0 equals to [(N / 2 + 1) / 2].

2.1.3 Quantization and Rate Distortion Control

The major purpose of quantization stage is to make a best tradeoff between available bits and acceptable distortion in each coding frame. The number of available bits depends on sampling frequency and the desired data rates. To increase the rate-distortion performance of audio encoding, the perceptual distortion levels for each block are derived by PAM. To maximize perceptual audio quality at a given bit budget, a strategy using two nested iteration loops [6] is employed to realize the rate distortion (R-D) control process.

A. Scalefactor bands

To reflect the characteristics of critical bands from PAM, the AAC system splits the spectrum data into sub-groups that are very close to the bandwidth of critical bands. At the sampling rate of 44.1 kHz or 48 kHz, there are 49 scalefactor bands for long blocks and 14 scalefactor bands for short blocks. For coding the scalefactor bands, a variable *sf_decoder* implies the quantizer step-size.

B. Non-uniform quantization

The AAC system uses the non-uniform companding quantization. The non-uniform

quantization applies coarse quantization steps for the MDCT coefficient of larger magnitudes and fine quantization steps to the MDCT coefficients of smaller magnitudes [15], which can control the overall distortion level of each block based on the masking thresholds derived by PAM. The signal to masking ratio (SNR) by the non-uniform quantization scheme remains constant with a wider range of MDCT coefficient energy as compared to the uniform quantization scheme.

At the AAC coding specification, the non-uniform quantizer is defined by

$$x\_quant = \text{int}\left(\left(\left| mdct\_line \right| \times 2^{-\frac{1}{4}(sf\_decoder - SF\_OFFSET)} \right)^{\frac{3}{4}} + MAGIC\_NUMBER \right) \tag{2}$$

and the inverse quantizer is defined by

$$x\_rescale = \left| x\_quant \right|^{\frac{4}{3}} \times 2^{0.25(sf\_decoder - SF\_OFFSET)}, \tag{3}$$

where *mdct_line* is the MDCT coefficient. *x_quant* is the quantized value. *x_rescale* is the reconstructed spectral value. *MAGIC_NUMBER* is defined to 0.4054 and *SF_OFSSET* is set as 100. The operator 'int' means to cast the remainder. *sf_decoder* implies the quantization stepsize.

The variable *sf_decoder* is split into two parts in the two nested iteration loops by

$$sf\_decoder = common\_scalefac - scalefactor + SF\_OFFSET, \tag{4}$$

where *common_scalefac* represents the bitrate control variable and *scalefactor* represents the noise shaping variable.

C. Bit reservoir control

The average number of available bits is calculated before doing the nested iteration loops. After the current frame is coded with the bits smaller than the available bits, the remaining bits are saved to the reservoir for the coding of next coding frame. The available bits for the next coding frame are computed with the number of bits decided by PAM (*more_bits*) and the unused bits (*bitres_bits*) stored at the bit reservoir.

The two nested iteration loops in the AAC coding standard consist of the inner loop and the outer loop. As shown in Figure 2-6, the inner loop can be considered as a rate control loop, which is devoted to change the quantizer step-size until the spectrum data can be coded with the given number of available bits. In addition, the quantization step-size is represented by *common_scalefac* with an initial value that equals to the lower bound of *common_scalefac*. With the quantization step-size and the target bits, the inner loop starts to process the MDCT coefficients and the real number of bits used per frame are counted by the Noiseless Coding module. The bit rate is controlled by increasing the value of *common_scalefac* until the number of used bits is less than or equal to the bit budget.

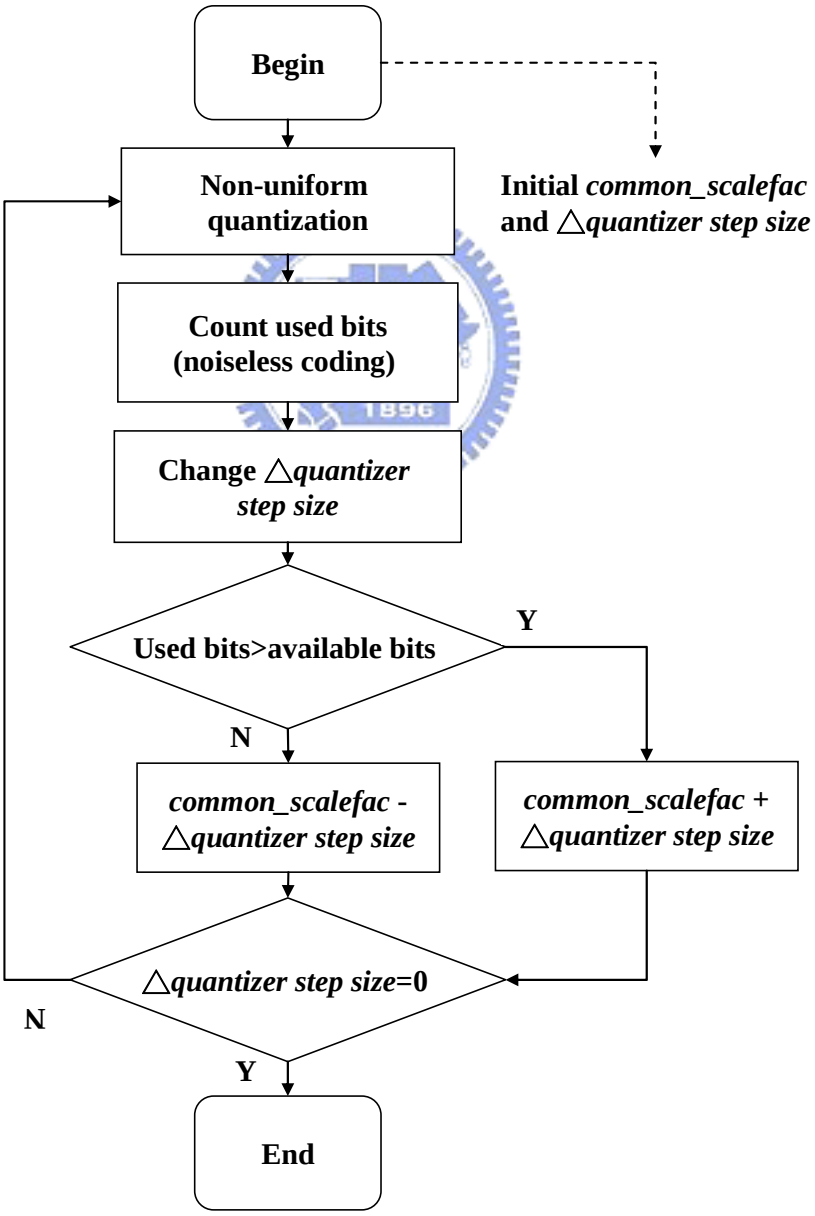

Figure 2-6. Block diagram of AAC inner loop iteration

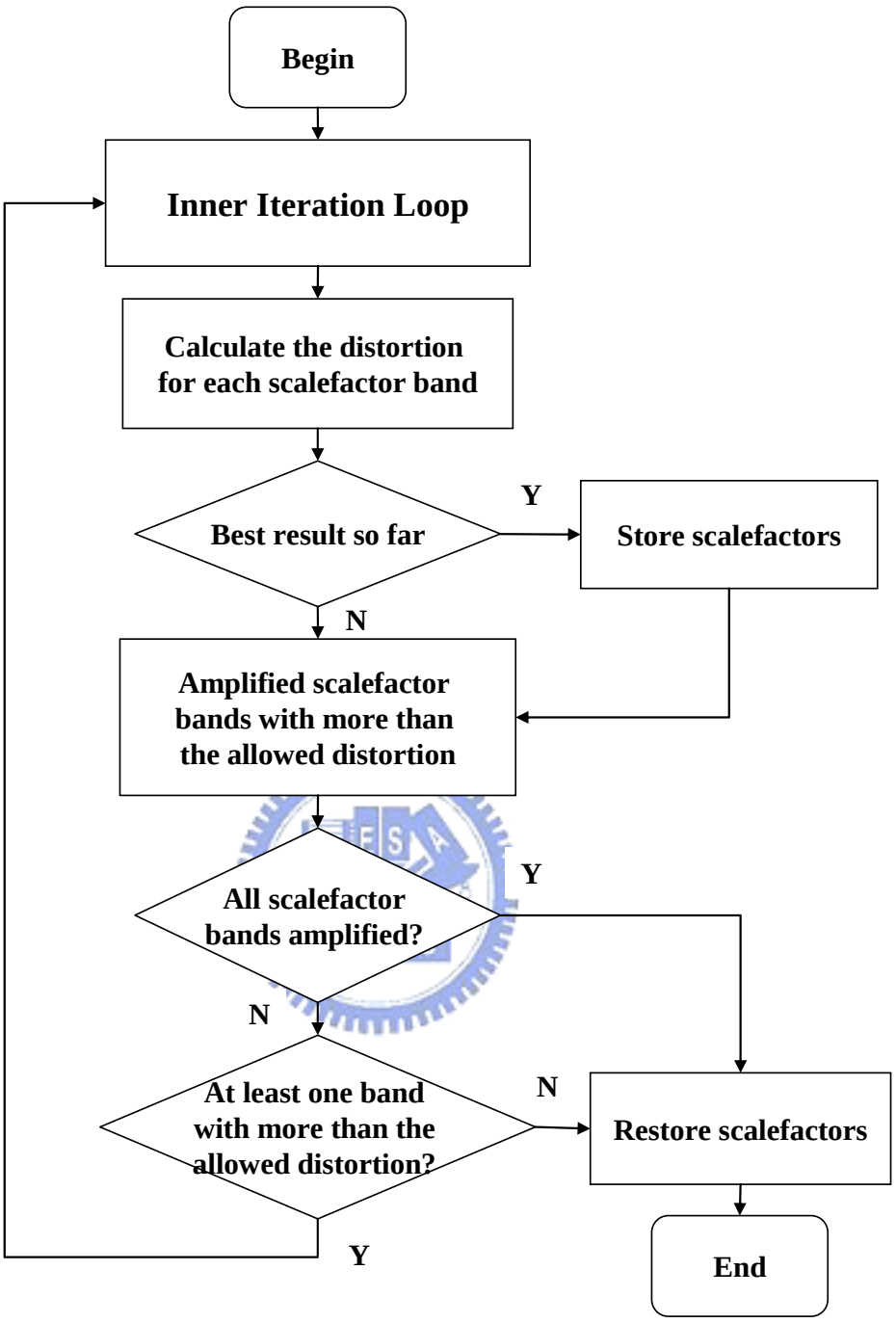

Figure 2-7. Block diagram of AAC outer loop iteration

The inner loop controls the bit rate per frame and the outer loop iteration focuses on maximize the audio quality according to the masking thresholds from PAM. Figure 2-7 shows the block diagram of outer loop iteration. First, the outer loop calls the inner loop and gets a intermediate quantized value that fits the bitrate requirement. Then, the error energy of reconstructed coefficients of intermediate quantized coefficient is calculated. For each

scalefactor band, the scalefactor is amplified when the error energy of reconstructed signal exceeds the allowed distortion level. The iteration stops as the minimum distortion is found at the given bit rate and the scalefactors are stored for the noiseless coding. When the process can not find the proper distortion level for the given bits, the iterations are enforced to terminate. The termination conditions, for example, occur when all the scalefactor bands are amplified or the difference between two consecutive scalefactors is greater than 60.

2.1.4 Noiseless Coding

The AAC noiseless coding stage that removes the statistical redundancy based on coefficients sectioning and Huffman coding (entropy coding) can efficiently encode the quantized spectral coefficients without loss of information.

The Noiseless Coding module allocates 1024 quantized coefficients into varying sections. For each section, a unique Huffman codebook that fits the statistical property of quantization coefficients at the section is used. Huffman coding is used to represent *N*-tuples (4 or 2 in AAC) of quantized coefficients. The appropriate codebook is chosen from 11 AAC Huffman codebooks, which are represented by the maximum absolute value of quantized coefficients. Each Huffman codebook has two classes of codewords that fit to the distinct signal probability distributions. The codebook 11 uses a special "escape coding" mechanism to encode the section with maximum quantized value that is equal or greater than 16. Huffman codebook zero requiring no codewords is applied to the sections that contain only zero coefficients.

The section boundaries can simply be set as the scalefactor band boundaries for better coding efficiency. To achieve the maximal rate-distortion performance, a greedy merging algorithm is used to minimize the total number of coding bits. In addition, to further increase the coding efficiency for the audio frames that employ the eight short windows, grouping and interleaving methods are enabled. The coefficients associated with continuous short windows can be grouped to share one set of scalefactors and the coefficients within each group are interleaved by reordering the scalefactor bands from each grouped short window.

In ACC coding systems, the final scalefactors are normalized by the global gain. The global gain is coded as an 8-bit unsigned integer and the scalefactors are differentially

encoded relative to the previous scalefactor.

2.1.5 Temporal Noise Shaping

Temporal Noise Shaping tool (TNS) is used to control the temporal shape of the quantization noise. The temporal noise shaping can be applied up to the block level of a filter bank. In the temporal domain, the quantization noise is not easily masked especially when the coded signals are “transient” or “pitch-based”. As mentioned in the psychoacoustic model, “pre-echo” happens since the pre-masking time is too short to mask out the noise.

TNS employs the duality characteristic between the time domain and the frequency domain to adjust predictive coding techniques. The adjustment implies if the signal is pulse-like, to increase coding performance, the signal is processed by either the direct coding in the time domain or the predictive coding in the frequency domain. Figure 2-8 illustrates that the spectral coefficients are sent to the TNS tool with an open-loop filtering. Based on TNS, some of the spectral coefficients are replaced with the prediction residual. Thus, the TNS tool can considerably enhance the audio quality for the speech and transient signals.

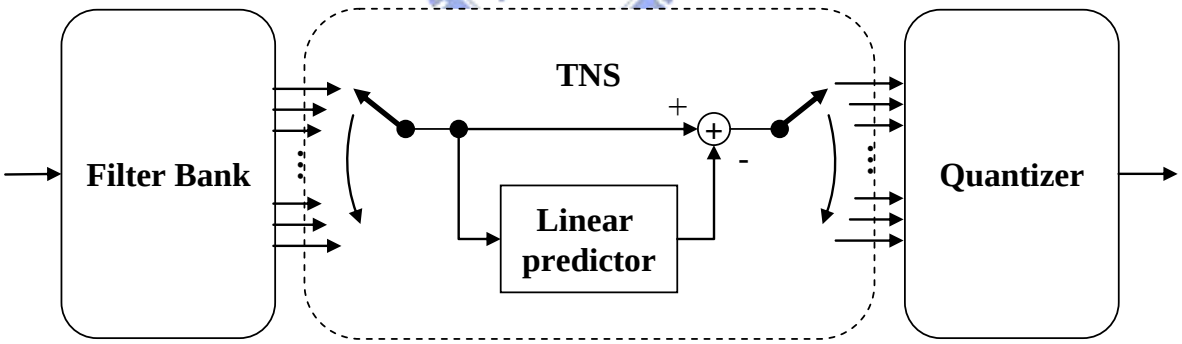

Figure 2-8. Block diagram of TNS filtering in an AAC encoder

2.1.6 Joint Stereo Coding

AAC joint stereo coding encodes stereo or multi-channel signals based on the cross correlation among the samples of different channels. By removing the redundancy, AAC joint stereo coding can reduce the total number of bits to encode the samples of different channels

separately. To optimize the coding efficiency, two different joint stereo methods defined in AAC standard can be selected to encode frequency bands.

A. M/S stereo coding

M/S coding that adopts a summation (M) channel and a difference (S) channel instead of left and right channels is very efficient for near monophonic signals whose differential signals energy is small. In addition, AAC coding system can decide to use M/S coding or left/right (L/R) coding at every noiseless coding band for all spectral coefficients within the processing block. When the signals at the left and right channels are highly correlated in statistic, the samples of two channels can be summed up to save the required bits by M/S stereo coding. Thus, when the statistical correlation of the samples from the different channels is higher than a specified threshold, the M/S stereo coding is applied. Otherwise, the L/R coding is applied. By choosing the better coding method to each block of audio samples, the AAC coding system can increase the overall coding performance.

B. Intensity stereo coding

The intensity stereo coding tool is used to exploit irrelevance between the intensities of high frequency signals for each pair of channels. For irrelevance reduction, the intensity stereo coding sums up the high frequency samples from the left and right channels. The magnitudes of summed samples are then rescaled by a specified factor. The rescaled signals can replace the corresponding high frequency samples at the left channel and corresponding signals at the right channel are set to zero. By removing the intensity irrelevance, the intensity stereo coding can improve the efficiency of AAC coding system.

2.2 Overview of Audio Bitrate Transcoders

In order to deliver audio signals over heterogeneous networks and to the devices with varying capabilities, the bitrate adaptation coding schemes are required. With a unitary bitstream of single audio sequence, the bitrate scalability methods are applied to match the network conditions and clients' capacities. For scalable audio streaming, scalable audio coders like MPEG-4 BSAC need side information. Therefore, scalable coding schemes may decrease overall coding efficiency as compared with non-scalable coding schemes at specified bit rates.

Thus from the rate-distortion performance viewpoint, the non-scalable audio coding schemes are introduced to realize the bitrate adaptation based on transcoding techniques. The transcoders can directly convert the compressed bitstream from higher bitrates to lower bitrates [18]-[20]. The bitrate transcoding approaches including a cascaded transcoder and existing rapid algorithms for bitrate scaling are introduced [18], [19].

2.2.1 Cascaded Transcoder

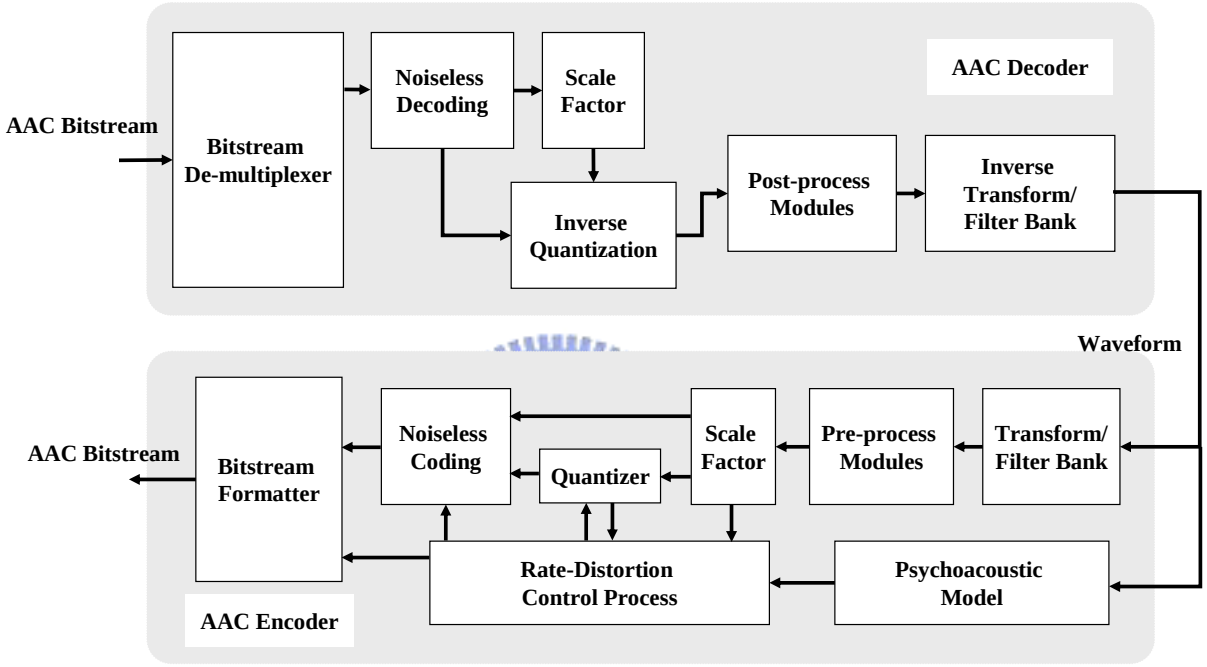

Figure 2-9. Block diagram of a cascaded transcoder.

In Figure 2-9, a transcoder cascades an AAC decoder and an AAC encoder. The cascaded transcoder could achieve the audio quality as AAC single layer coding. In the cascaded transcoder, the decoder reconstructs the original compressed bitstream with a specific bitrate to audio waveform samples and the encoder re-compresses the audio samples into a new bitstream of a specified bitrate. Thus, using the cascaded transcoder, the bitstream is converted to the audio samples in the time domain and the time-domain signals are transformed back to the spectral coefficients in the frequency domain for re-encoding, which takes a huge amount of computation. The high computation complexity of the cascaded transcoder will decrease the effectiveness of audio streaming applications. Thus, the key issue is to reduce the complexity of the most computational burden modules with a audio coding

system.

Figure 2-10 shows the computational time between the modules of cascaded transcoder. The test platform is based on free software faac and faad [17], and the experiment results are averaged from five stereo sequences from EBU [21]. The results show that the psychoacoustic model and R-D control process in AAC encoding process are the most computational burden modules. Thus, to speed up the audio transcoding, some fast algorithms reduced the computation complexity of the psychoacoustic model and/or R-D control process to save the power consumption for audio streaming.

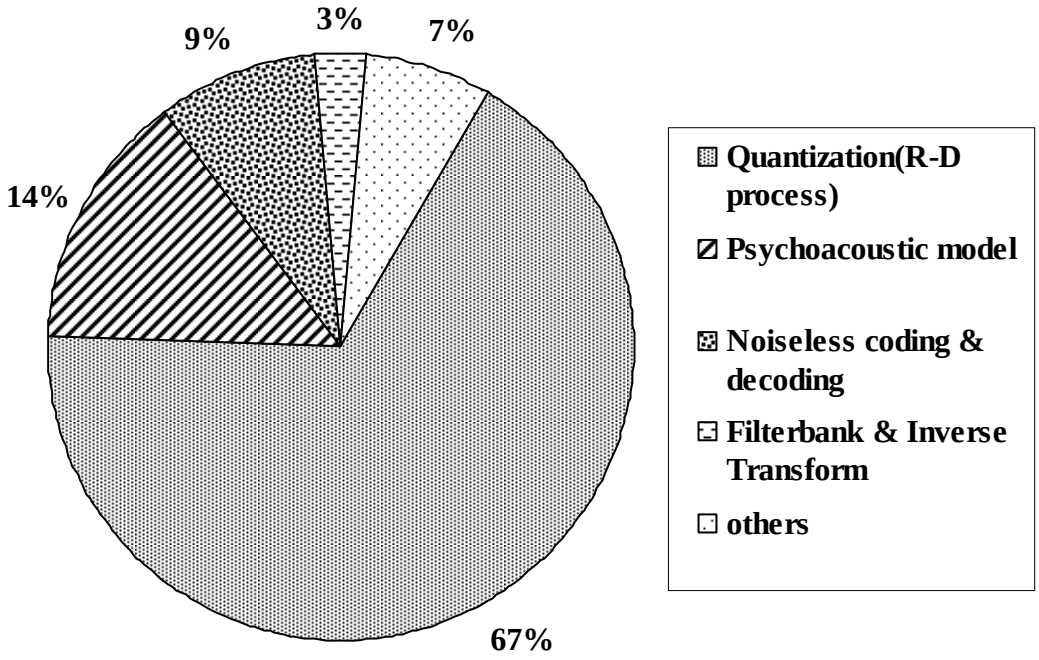

Figure 2-10. Percentages of function time at an AAC cascaded transcoder

2.2.2 Transform Domain Transcoder

There are three bitrate scaling algorithms that convert the bitstreams with the reconstructed spectral coefficients [19]. The conversion algorithms consist of bandwidth limitation, re-quantization and perceptual re-quantization.

A. Bandwidth limitation

Audio information of lower frequency bands is more important than that of higher

frequency bands to reconstruct the audio samples. Based on the information importance, the transcoding techniques with bandwidth limitation can reduce the bit rate by allocating zero bit to the high frequency bands. As shown in Figure 2-11, the zero bit allocation is applied from the highest frequency band to the lowest frequency band until the bitrate is less than or equal to the target bitrate.

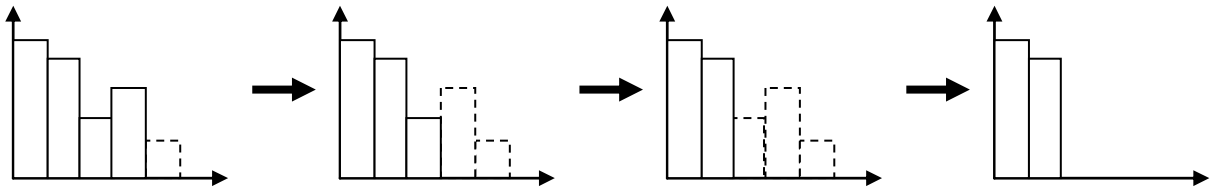

Figure 2-11. Bandwidth limitation

B. Re-quantization

At the transcoders with re-quantization, the bit allocation is done by iteratively subtracting one bit of frequency bands every frame. As shown in Figure 2-12, the bit subtraction is applied from the highest frequency band to the lowest frequency band until the required bitrate is less than or equal to the target bitrate for the processing frame.

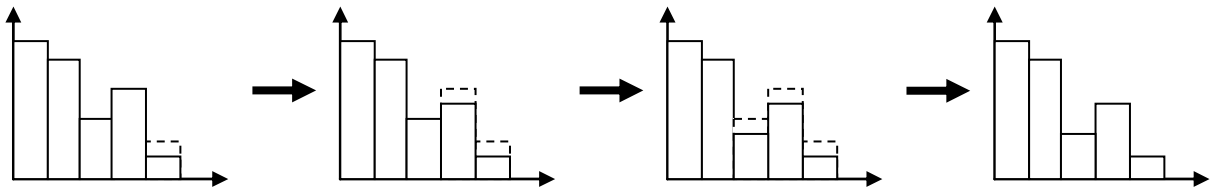

Figure 2-12. Re-quantization

C. Re-quantization based on the psychoacoustic masking thresholds

In the beginning of transcoding, all NMR values of each frequency band are set to zero. The quantization step-size is increased by one for each frequency band that will contribute the least quantity of NMR. As shown in Figure 2-13, the process is repeated until the required bitrate is less than or equal to the target bitrate of each frame.

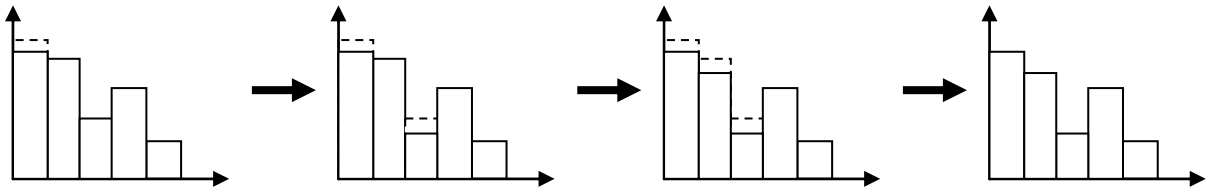

Figure 2-13. Re-quantization reflecting psychoacoustic model

The experimental results showed that the transcoding algorithm using the perceptual masking re-quantization has better quality than the other two algorithms. The big challenge for the third transcoder is the masking thresholds are derived with the reconstructed audio samples at the psychoacoustic models that are built with the uncompressed audio samples.

2.2.3 Single Layer AAC Transcoder (SLAT)

In [18], the fast single layer AAC transcoder manipulated the transcoded bitstreams in the "quantized spectrum domain". The bitstream manipulation is based on based on the linear relationship of the required bits and the percentage of nonzero quantized spectral coefficients. By modeling the linear relationship and reusing the information within the original bitstreams, SLAT can speed up the cascaded transcoder by replacing the nested loop at the bit reservoir control with a linear prediction. Thus, SLAT can save the time taken by the forward and inverse filter bank transformation, the psychoacoustic model and the R-D control process, which are the most computational burden modules of an AAC coding system. In addition, SLAT can retain the coding performance close to the cascaded method.

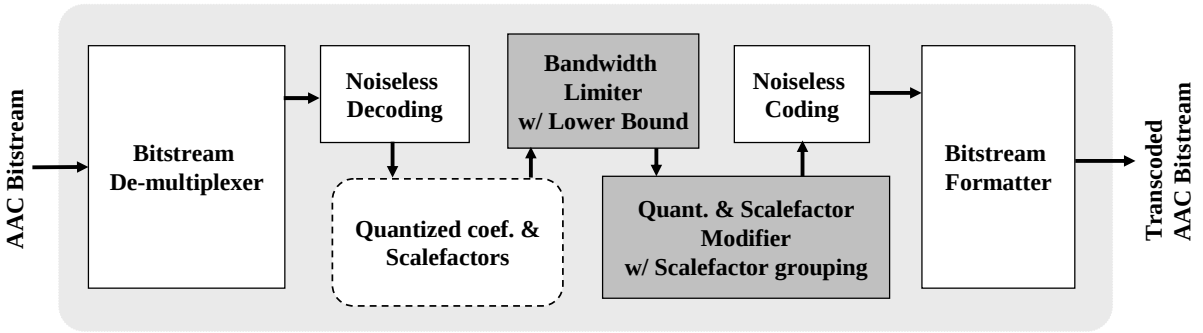

Figure 2-14. Block diagram of SLAT

Figure 2-14 shows the architecture of SLAT that manipulates the transcoded bitstreams

in the “quantized spectrum domain” in order to maximize the overall transcoding throughput. After the AAC input bitstream is decoded by the Noiseless Decoding module, the new bitstream is generated with the quantized spectral coefficients. A set of quantized spectral coefficients and scalefactors (*sf_decoder* in Eq.(2)) are modified by three bitrate reduction techniques, which mean the bandwidth reduction with the ρ-domain model, bit reduction for encoding the lower frequency coefficients and side information reduction. Figure 2-15 shows the combination of three reduction methods in the SLAT architecture.

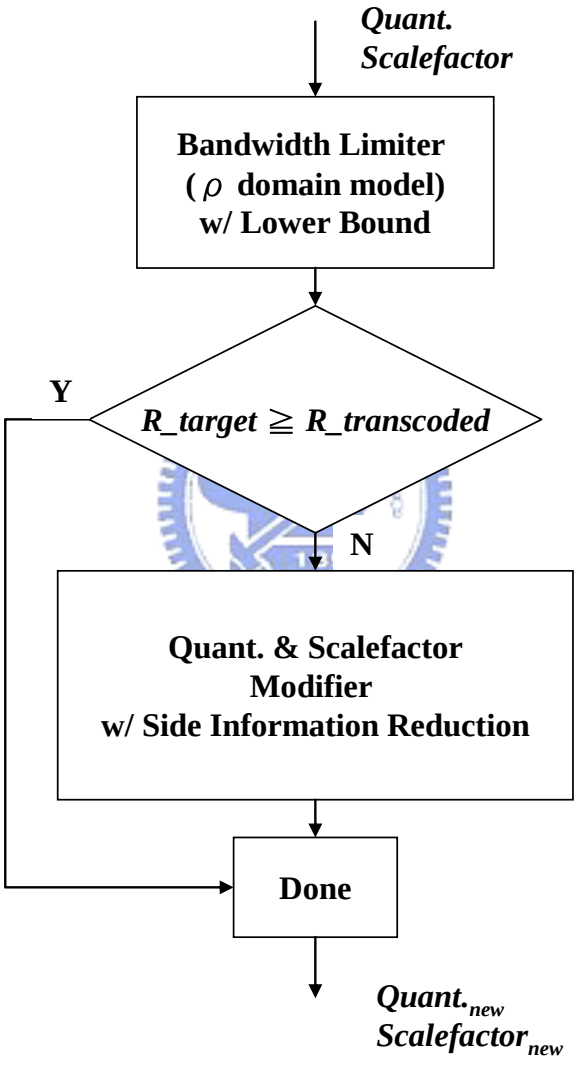

Figure 2-15. Algorithm flow chart of SLAT

A. Bandwidth reduction with the ρ-domain model

The approximated linear model between the coding bitrate and percentage (ρ) of non-zero quantized coefficients is used to limit the bandwidth for lower bitrate bitstreams.

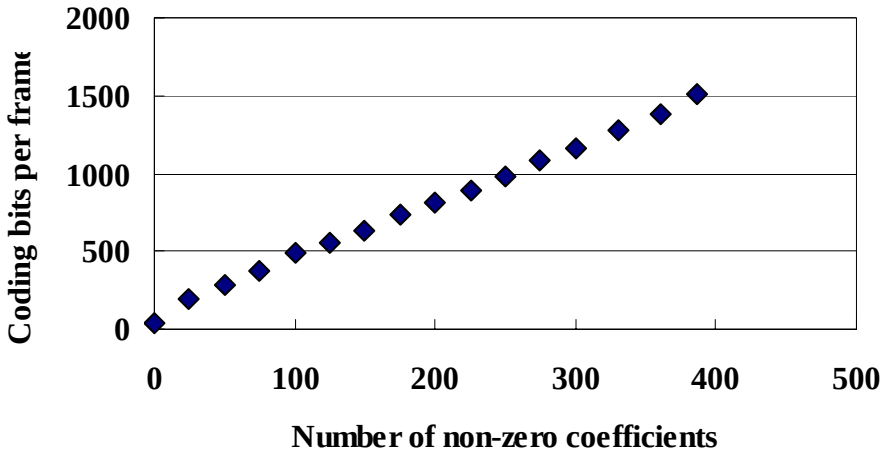

Figure 2-16. Linear model between the coding bitrate and the percentage (ρ) of non-zero quantized coefficients.

Figure 2-16 demonstrated the ρ-domain model in an AAC coding frame. In Eq. (5), the approximated linear relationship is used to estimate the total number of non-zero coefficients to be discarded from the higher frequency bands.

$$\frac{R_o}{N_o} \cong \frac{R_t}{N_t}, \tag{5}$$

where R_o is the original bitrate, R_t is the target bitrate, N_o is the non-zero coefficients of original bitstream and N_t is the predicted non-zero coefficients for the target bitrate.

The difference between N_o and N_t represents the total number of non-zero coefficients to zero out. The operation to set the coefficients to zero is applied from the highest frequency band. The removal of the high frequency bands may decrease the listening quality. To retain the quality, the lowest bounds to remove the high frequency bands at different target bitrates should be set.

B. Bit reduction for encoding the lower frequency coefficients

To save the bits to encode the lower frequency coefficients, the R-D relationship in quantized spectrum domain is analyzed. We proved that both the bitrate and distortion of transcoded bitstream can be formulated as a function of "quantized coefficient of original bitstream" and "increase of *sf_decoder*". In addition, the observations on the rate-distortion

curve showed that the maximum distortion takes place when the quantized coefficients are set from the unity to zero. The bit reduction changes the magnitude of the quantized coefficients by increasing *sf_decoder* with the average quantized value. To retain the listening quality, the nonzero quantization coefficients of the original bitstream shall have the values larger than the unity.

In the beginning of bit reduction, the quantized coefficients are averaged in a scalefactor band. In Eq.(6), q_i represents the *i*-th quantized coefficient. $q_{avg,b}$ and sf_length_b denote the averaged value and the length of the *b*-th scalefactor band respectively.

$$q_{avg,b} = \frac{\sum_b q_i}{sf\_length_b}. \tag{6}$$

The scalefactor difference $sfd_{one,b}$ is calculated and the average quantized value is diminished to the unity by

$$1.0 = q_{avg,b} \cdot 2^{-\frac{3}{16} sfd_{one,b}}. \tag{7}$$

Given $sfd_{one,b}$, the quantized value $q_{new,i}$ is calculateby

$$q_{new,i} = \text{int}\left(q_i \cdot 2^{-\frac{3}{16} sfd_{one,b}} + 0.4054 \right). \tag{8}$$

Based on the re-quantized values, the bits $Bit_{new,b}$ needed for the scalefactor band *b* is calculated by Huffman coding. $Bd_{one,b}$ is the difference between the original bits $Bit_{ori,b}$ and the current bits $Bit_{new,b}$.

$$Bd_{one,b} = Bit_{ori,b} - Bit_{new,b}. \tag{9}$$

A $ratio_b$ is calculated in Eq.(10), which represents the number of bits can be reduced by increasing the scalefactor by one step with an averaged quantized value that is low bounded to

the unity.

$$ratio_b = \frac{Bd_{one}}{sfd_{one\ b}}. \quad (10)$$

The increase of the estimated scalefactors for the entire frame sfd_{frame} is calculated in Eq. (11). Bd_{frame} is the number of bits to reduce for the current frame.

$$. \ sfd_{frame} = \frac{Bd_{frame}}{\sum_{frame} ratio_b}. \quad (11)$$

Thus, the scalefactors can be updated by adding sfd_{frame} to the original scalefactor. The quantized coefficients are updated by replacing $sfd_{one,b}$ with sfd_{frame} in Eq. (8).

C. Side information reduction

The side information in AAC occupies a high percentage of coding bits at low bitrates. To reduce the bits of side information, SLAT decreased the difference of successive scalefactors and set zero codebook for the zero quantized coefficients. The experiment results showed that noise-to-masking ratio (NMR) degradation by SLAT is less than 1.0 dB compared to the cascaded transcoder. SLAT can speed up the cascaded transcoder by 5 times.

2.2.4 Summary

The cascaded transcoder can get listening quality as an AAC single layer coding system. Since the input signal of the transcoder is a lossy compressed bitstream, which means the transcoded quality is bounded by the input bitstream at the original bitrate. In addition, the cascaded transcoder that inherits all modules from the AAC single layer coding system is computationally expensive.

The re-quantization reflecting psychoacoustic model [19] can obtain the best transcoding quality when the uncompressed audio sequences are used as the input to psychoacoustic model. As the input to the psychoacoustic model within the transcoder is the reconstructed audio signals, the masking thresholds are not accurate to prevent the audible distortion. In

addition, the re-quantizing process that iteratively decreases the allocated bits may still cost lots of computation time.

SLAT [18] can retain the listening quality with rapid transcoding algorithms. The three combined algorithms still have a few implementation problems. In the beginning, the bandwidth is reduced with the ρ-domain model, but the low bound of bandwidth reduction is difficult to set at different target bitrates. In Eq. (11), the scalefactor increment is in proportion to the bits to reduce. The remaining bits after the bandwidth reduction should be reduced by the "Quant & Scalefactor Modifier" to reach the target bitrate. To meet the target bitrate, the "Quant & Scalefactor Modifier" will re-quantize the coefficients to zero when the number of bits to reduce is greater than a specified threshold after the bandwidth reduction. The re-quantization by the "Quant & Scalefactor Modifier" violated the original hypothesis that the minimum re-quantized coefficients should be bounded to the unity. The violation avoids the transcoding from converging to the target bitrate rapidly.

Chapter 3

Fast Rate-Distortion Optimized Transcoder

This chapter describes a fast and optimized transcoding algorithm under the criterion of minimal noise-to-masking ratio (NMR). The transcoding algorithm is called as Fast Rate-Distortion Optimized Transcoder (FRDOT). The transcoding architecture of FRDOT is referred to Single Layer AAC Transcoder (SLAT). To further improve the transcoding performance at low bitrates, FRDOT presents a new algorithm to optimize the listening quality in NMR for transcoded audio bitstreams. In addition, a bitrate control module (BCM) is proposed to meet with the given bit budget of the processing bitstream.

3.1 Architecture of FRDOT

For rate distortion optimized content adaptation, transcoding techniques must support an efficient bitrate reduction with optimal audible quality under the criterion of minimal noise-to-masking ratio (NMR). In addition, to fit to the real-time applications, the transcoders of lowest complexity and small memory bandwidth are strongly demanded. To simultaneously address the transcoding issues including the minimal NMR and the lowest complexity, we present a NMR optimized transcoder, which is called the Fast Rate-Distortion Optimized Transcoder (FRDOT), based on the linear model between the coding bitrate and the percentage (ρ) of non-zero quantized coefficients [18].

Figure 3-1 shows the architecture of FRDOT transcoder. The architecture is similar to the architecture of SLAT [18]. Within FRDOT, to address the high complexity issues of the cascaded transcoder, the time-consuming modules like PAM and the quantization are removed. In addition, to match the allocated bits per frame and the target bitrate, a bitrate control module (BCM) is used within FRDOT to estimate the total number of bits to reduce for each coding frame at a given bitrate. In addition, for improving the rate-distortion performance, the NMR optimized algorithm with a bandwidth limiter is presented. The bandwidth limiter iteratively discards the quantized coefficients from the highest scalefactor band to the lowest

frequency band. The removal of the quantized coefficients is terminated when the total number of consumed bits is smaller than or equal to the specified bit budget. In summary, FRDOT consists of the BCM, the NMR optimized algorithm and the bandwidth limiter.

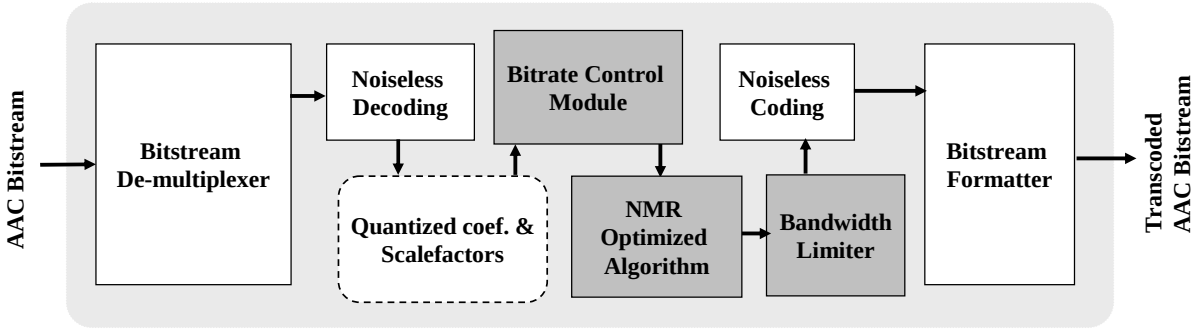

Figure 3-1. Block diagram of the FRDOT transcoder.

3.1.1 Bitrate Control Module (BCM)

BCM can estimate the total number of bits to reduce in the current audio frame as the target bitrate of transcoding bitstream is lowered. The estimation for each scalefactor band is based on two factors. One is the difference of the target bitrate and the original bitrate and the other is the total amount of bits that are reduced at the previous scalefactor bands. Therefore, to reduce the bits precisely, the estimation can be dynamically adjusted in a manner of frame by frame based on inter frame relationship. Based on the estimation of bit reduction, the flow chart of BCM is shown in Figure 3-2.

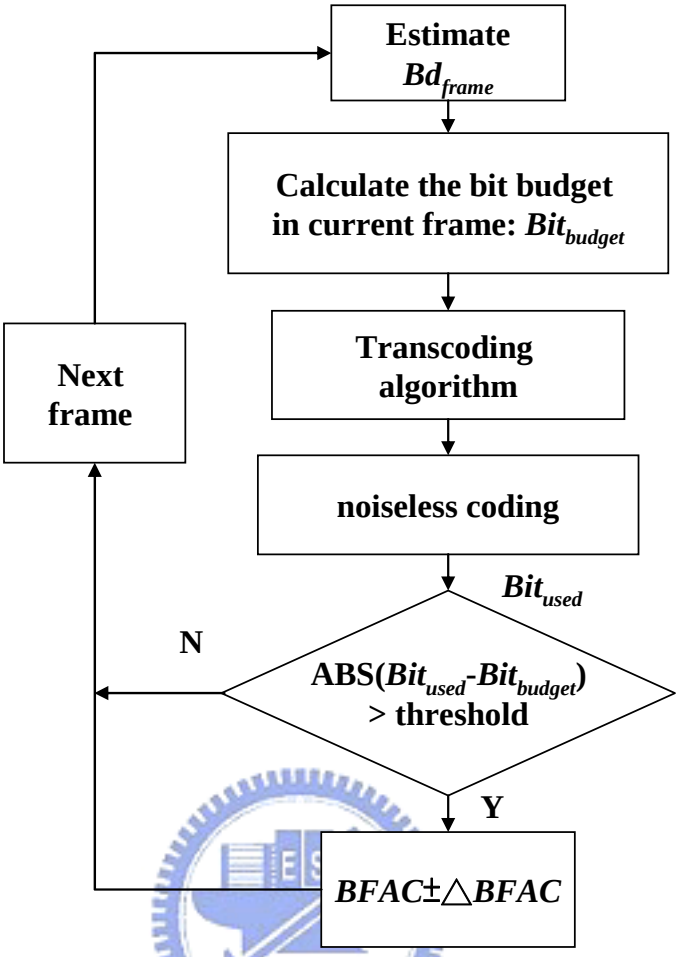

Figure 3-2. Flow chart of Bitrate Control Module (BCM)

For the *n*-th frame, the total number of bits to reduce at a given bitrate is estimated by

$$Bd_{frame,n} = Bd_{frame,n-1} + \Delta Bit_{n-1}, \tag{12}$$

where $Bd_{frame,n}$ and $Bd_{frame,n-1}$ mean the amount of the bits to remove from the *n*-th and the *(n-1)*-th frames respectively. ΔBit_{b-1} presents the amount of the bits not able to remove at the *(n-1)*-th frame.

The number of reduction bits in the *n*-th frame is evaluated by multiplying *BFAC* with the difference between the original bitrate and the target bitrate.

$$Bd_{frame,n} = (R_{ori} - R_{tar}) \times BFAC, \tag{13}$$

where R_{ori} indicates the original bitrate and R_{tar} means the target bitrate. *BFAC* represents a ratio between the difference of coding bits except the bits to encoder side information and the

difference of original and target bitrates.

$$BFAC = \frac{Bit_{ori} - Bit_{tar}}{R_{ori} - R_{tar}}. \tag{14}$$

The observation showed that the magnitudes of *BFAC* are in a small rage of [9.3..9.8]. In FRDOT, we set the initial value of *BFAC* to be 9.5. When the difference of the used bits and left bits of the previous frame is larger than a predefined threshold, *BFAC* is updated by Eq.(15).

$$BFAC \pm \Delta BFAC, \text{ if } \left| Bit_{used,b} - Bit_{budget} \right| > threshold. \tag{15}$$

When the number of used bits exceeds the number of the available bits by a constant threshold that is found empirically, the bit reduction for the processing frame is underestimated. To fix the underestimation, a delta value is added to *BFAC*, which can provide a more precise estimation for the next frame. To avoid the bit reduction from the overestimation, *BFAC* is subtracted by the delta value.

The bits ΔBit_{b-1} that are not reduced after fully encoding the *(n-1)*-th frame by the noiseless coding are derived by

$$\Delta Bit_{n-1} = Bit_{used,n-1} - Bit_{budget}, \tag{16}$$

where $Bit_{used,n-1}$ means the number of used bits at the *(n-1)*-th frame and Bit_{budget} represents the bit budget of the current frame. Thus, by Eqs. (13) and (16), Eq. (12) is changed to

$$Bd_{frame,b} = (R_{ori} - R_{tar}) \times BFAC + (Bit_{used,n-1} - Bit_{budget}), \tag{17}$$

Finally, the available number of bits allocated to the current frame is calculated at a specified bitrate.

$$Bit_{budget} = R_{tar} \times 1024 / sampling\_rate \tag{18}$$

After the coding bits are reduced by transcoding algorithms, the actual number of bits used to encode the current frame is calculated by the noiseless coding.

3.1.2 Modified SLAT

The architecture of FRDOT is referred to the architecture of SLAT that reduces the transcoding complexity and memory based on the linear model [18]. To further address the issues of complexity reduction and rate-distortion performance for NMR optimized transcoding, the modified SLAT is described.

The major problem of the SLAT algorithm is to derive proper thresholds for bandwidth reduction. The improper thresholds will lower the performance of "Quant. and Scalefactor Modifier". When "Quant. and Scalefactor Modifier" considers a great amount of bits to reduce after the bandwidth reduction, the scalefactors will be increased and the reconstructed coefficients are zeroed out to meet the target bitrate. When the percentage of the zero coefficients after re-quantization is increased, the overall listening quality is decreased. Thus, the remaining issue is to control the total amount of the coefficients to zero out for retaining the listening quality at a given bitrate.

The number of the coefficients to zero out is estimated by Bitrate Control Module (BCM). With BCM to compute the number of reduction bits, the modified SLAT in Figure 3-3 can estimate the number of the bits to reduce first using the BCM. By Eq. (9), we sum up the bits of each scalefactor band to get the maximum number of bits that can be reduced by the "Quant. and Scalefactor Modifier". If the number of bits exceeds the maximum number of bits allocated to the current frame, the bandwidth reduction of SLAT is applied to discard high frequency bands. After the bandwidth reduction, the "Quant. & Scalefactor Modifier" will increase the scalefactor to address a precise bitrate reduction.

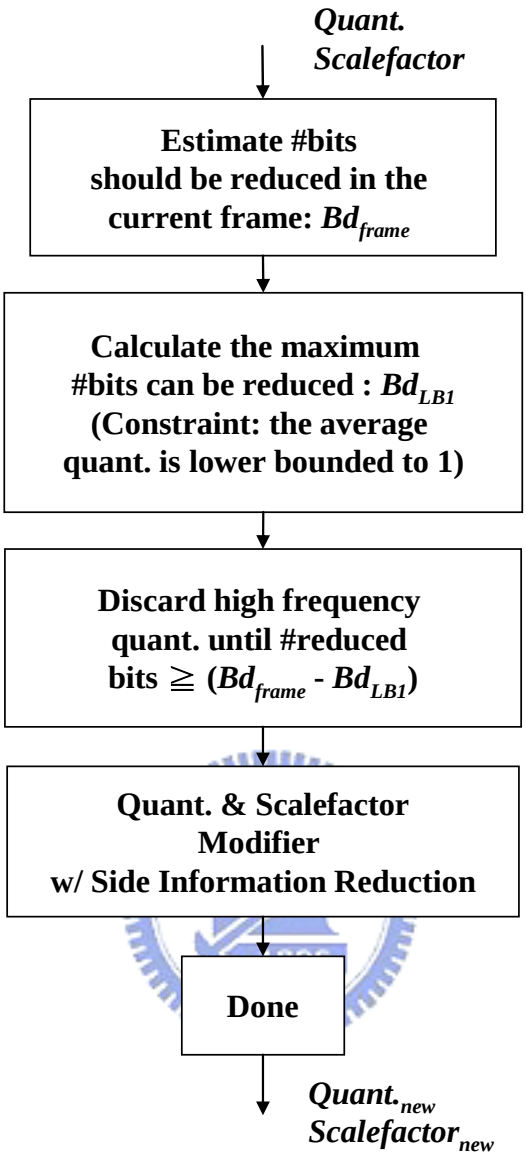

Figure 3-3. Flow chart of the modified SLAT

Figure 3-4 shows that modified SLAT with automatically-adjusted bandwidth reduction can improve the original SLAT with manually-tuned bandwidth reduction by 0.5 dB in NMR. With the enhanced bandwidth reduction and bit allocation, the listening quality by the modified SLAT is worse than the cascaded transcoder by 0.2 to 2.0 dB in NMR at the bitrates smaller than 80 kbps. Thus, we present a novel NMR-optimized AAC transcoder to improve the listening quality of modified SLAT with low complexity for a high to low bitrate conversion.

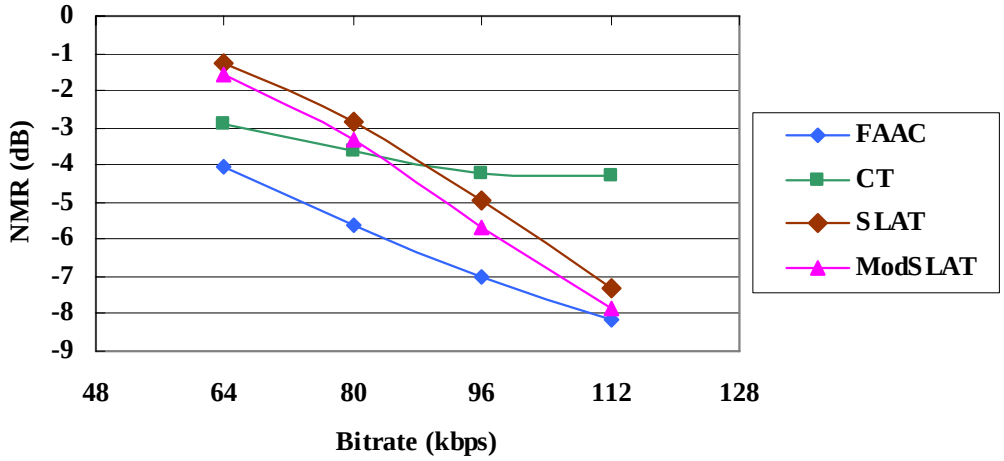

Figure 3-4. NMR comparison of the modified SLAT and the SLAT

3.2 NMR Optimized Transcoding Algorithm

For fast rate-distortion optimized content adaptation, the NMR optimized AAC transcoder will simultaneously address the issues including the minimal NMR, the low complexity, and the small memory bandwidth. The main goal of NMR optimized bitrate adaptation transcoding is to find the best scalefactor sfd_{best} for the best listening quality under the NMR criterion at a given bitrate. Thus, the core of NMR optimized algorithm rapidly searches for the best scalefactor to generate the transcoded bitstream with the maximal listening quality.

3.2.1 Rationale

The difference of scalefactors between the current band and the immediately previous band is called as scalefactor increment, which is denoted as *sfd*. The best scalefactor increment sfd_{best} is derived by subtracting the final scalefactor from the scalefactor of the original bitstream. The final scalefactor increment is encoded by lossless coding module as the side information for reconstruction. Therefore, to reduce the bits of side information, the range of scalefactor increments shall be limited. In addition, the final scalefactor increment will affect on the total amount of bits to encode each MDCT coefficient and the reconstructed magnitude of each audio sample. Thus, the rate distortion performance of rate adaptation transcoding will strongly depend on the scalefactor increment of each band.

To realize NMR optimized transcoding, the relationship between the scalefactor increment and the quantization error as shown in Figure 3-5 was investigated. With different scalefactor increments to quantize specified MDCT coefficients, the dotted line presents the levels of the quantized coefficient without rounding and the stair-like curve means the levels of the quantized input coefficients with rounding. The dashed area indicates the error occurred at the input coefficient with varying scalefactor increments.

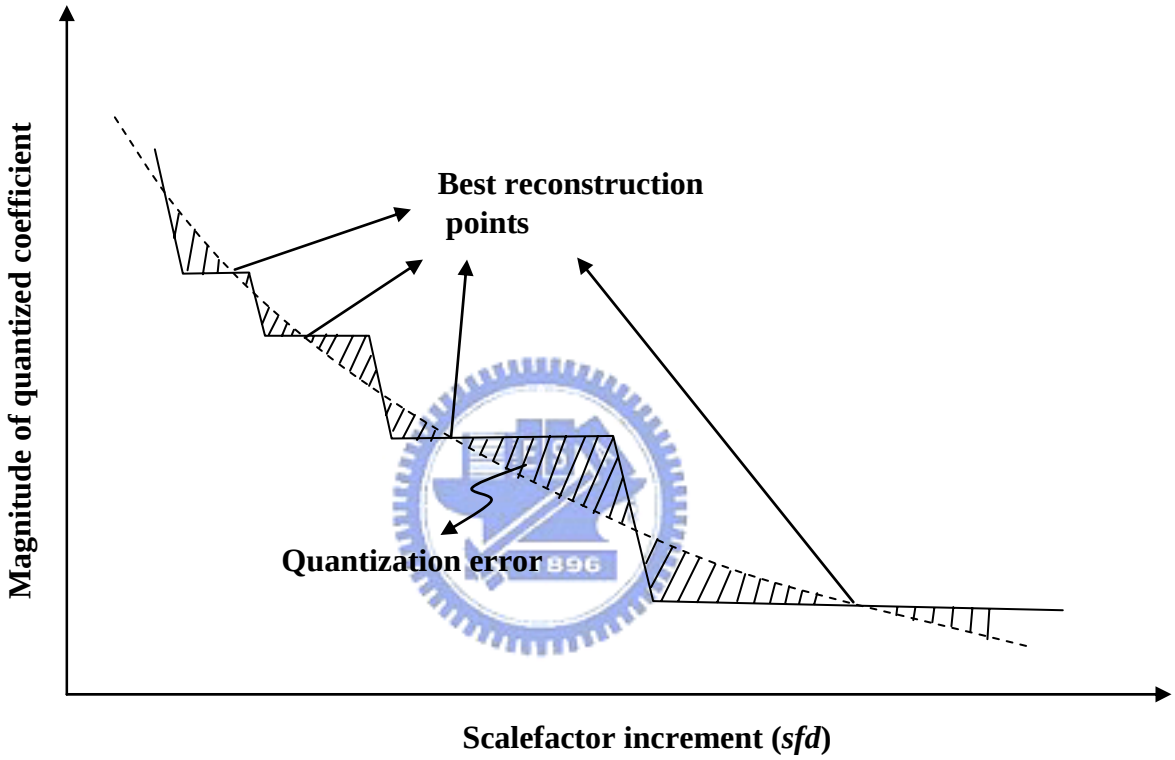

Figure 3-5. The illustration of the quantization error of an input coefficient.

As the scalefactor increment is enlarged from the minimum to the maximum, the levels of the quantized coefficients are descent. In addition, the levels of quantized coefficient are identical for at a small range of scalefactor increment. Therefore, for any input coefficient, applying a small set of scalefactor increments will generate the same level of the quantized coefficient, which is consistent with the observations in Figure 3-5 and Figure 3-6. The same levels of the quantized coefficient will take an identical number of bits into the transcoded bitstream after AAC lossless coding. At the decoder side, with different scalefactors for a unitary level of every quantized coefficient, the reconstructed values are varied, which

indicates the different quantization error and listening quality of the transcoded bitstreams. Thus, to realize the rate-distortion optimized (RDO) transcoder, we search for the best scalefactor increment (best reconstruction points) under the NMR criterion at a given bitrate

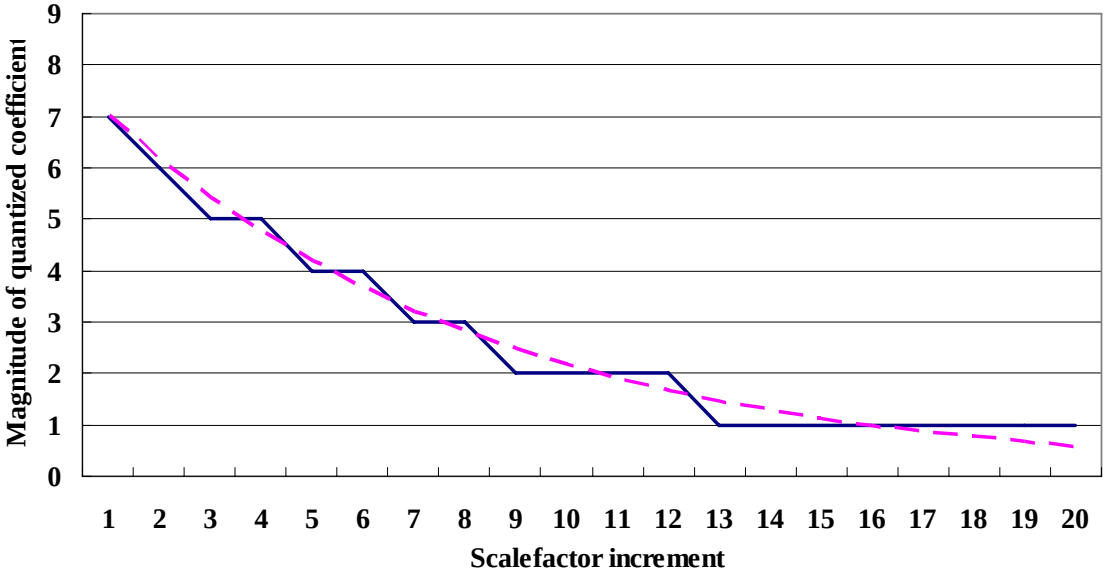

Figure 3-6. The relationship between the re-quantized coefficient and the scalefactor increment for the input coefficient q_i equal to 8.

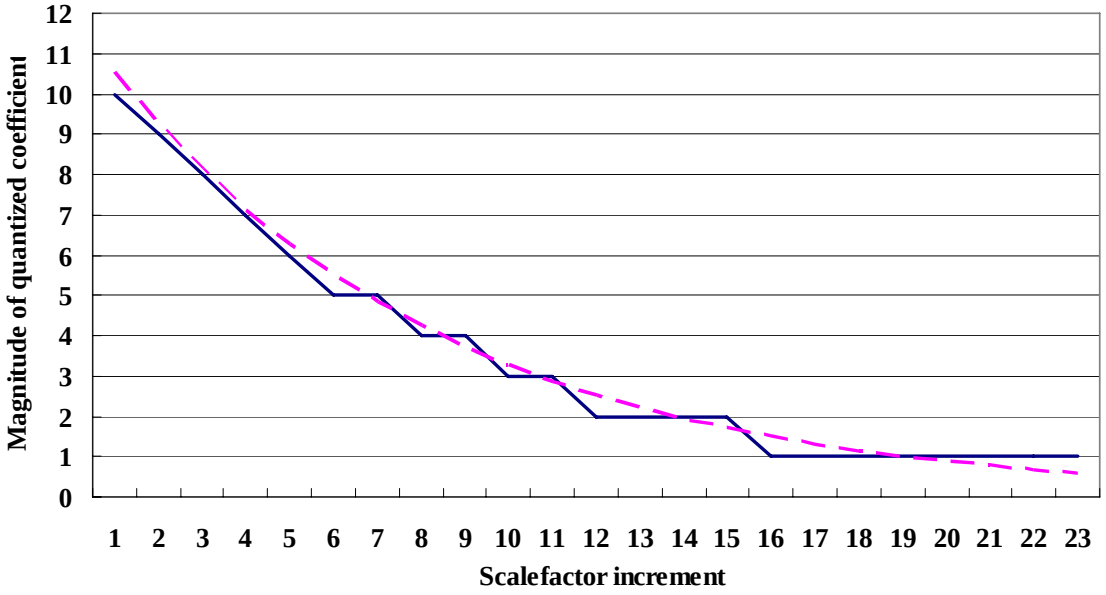

Figure 3-7. The relationship between the re-quantized coefficient and the scalefactor increment for the input coefficient q_i equal to 12.

3.2.2 NMR-Based Rate-Distortion Optimization

The NMR-based rate-distortion optimization is based on NMR-based search for the best scalefactor increment. To search for scalefactor increments under the NMR criterion, the transcoder must derive the masking thresholds based on the psychoacoustic model that is built with the uncompressed audio signals. With the compressed audio signals from the archived bitstreams, the masking thresholds can not be used for perceptual audio coding. In addition, derivation of the masking thresholds will take lots of computation cycles, which does not match the real-time transcoding requirement. Thus, we speed up the NMR optimized transcoding based on the embedded information within the input bitstreams.

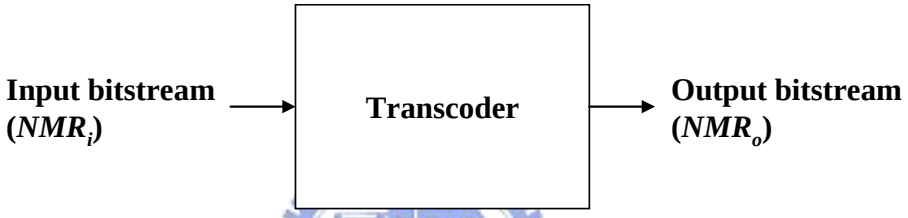

Figure 3-8. A sketch map of transcoder

Figure 3-8 illustrates a transcoder with an input bitstream and an output bitstream. The noise-to-masking ratios (NMR) of input and output bitstreams are denoted as NMR_i and NMR_o respectively. NMR_i is the upper bound of NMR_o, since conversion of the input bitstream to the output bitstream at a lower bitrate, the audio quality is degraded. The NMR degradation by the transcoding process can be formulated by

$$\begin{aligned} & NMR_{\text{deg}\,rade} \\ &= NMR_o - NMR_i \\ &= (SMR_o - SNR_o) - (SMR_i - SNR_i) \\ &= (SNR_i - SNR_o) \\ &= f(SNR_o), \end{aligned} \tag{19}$$

where SMR_o and SMR_i present the signal-to-masking ratios (SMR) of the audio signals at the output and input bitstreams respectively. SNR_o and SNR_i denote the signal-to-noise ratios (SNR) of the audio signals at the output and input bitstreams respectively.

The NMR value is to quantify the energy of audible noise. For the same audio source, the reconstructed audio signals with a smaller NMR value have better audio quality than the reconstructed audio signals with a larger NMR value. Therefore, Eq. (19) defines the NMR degradation by subtracting NMR_o with NMR_i. By definition, NMR can be represented by the SMR minus SNR. Thus, the NMR degradation equals to the difference of SNR values plus the difference of SMR values. The SNR value is measured with the audio source waveforms and the reconstructed audio waveforms. The SMR value is derived by the psychoacoustic model with the audio source signals. Thus, for the same audio source and the same psychoacoustic model, the SMR difference (ΔSMR) is set as zero and the NMR degradation can be formulated as a function of the SNR difference for bitrate adaptation transcoding. As the SNR_o approaches to SNR_i, the audible quality of reconstructed signals from the input and output bitstreams is close. Thus, the minimization of NMR degradation can be replaced with the minimization of SNR degradation at a given bitrate in bitrate conversion.

After deducing the optimization criterion from NMR to SNR, the schema of NMR optimized algorithm for each audio frame.

1. Given the difference of original and target bitrates, the number of bits to reduce at a scalefactor band of the handling frame is estimated. The bits allocated to each scalefactor band shall be large enough to make the averaged value of quantized coefficients greater than 1.
2. The scalefactor increment is estimated based on the difference of the bitrate of the original bitstream and the target bitrate of transcoded bitstream.
3. A specified search range to fine tune the scalefactor increment is defined.
4. To optimize the SNR value at the scalefactor band, an optimal increment of scalefactor is obtained by a full search within the predefined search range.
5. The best scalefactor for coding the current scalefactor bands equals to the summation of the scalefactor increment and the original scalefactor.
6. With the final scalefactor, the reconstructed coefficients are re-quantized and encoded into the transcoded bitstream.
7. The steps 1 to 6 are applied from high to low frequency scalefactor bands of the current frame.

For an AAC coding system, SNR of the output bitstreams can be derived from the

quantization formula. To further save the coding time, the quantized output coefficients are re-quantized without inverse quantization. Referring to Eq. (3) and Eq. (4), the quantization formula is defined as

$$q_i = \text{int}\left(x \cdot 2^{-\frac{3}{16}\cdot sf_i} + 0.4054 \right), \qquad x = \left| mdct\_line \right|^{\frac{3}{4}}, \tag{20}$$

where x is the compressed MDCT coefficient. q_i and sf_i represent the i-th quantized coefficient and the i-th scalefactor of input bitstream respectively. The operator 'int' means to cast the remainder. *mdct_line* is the MDCT coefficient.

When q_i is inversely quantized, a reconstructed value of the MDCT coefficient x_{Ri} is obtained by

$$x_{R_i} = q_i \cdot 2^{\frac{3}{16}\cdot sf_i} \tag{21}$$

When the bitrate of the input bitstream is reduced, the quantized coefficient is decreased from q_i to q_o and the scalefactor quantity is increased from sf_i to sf_o. The scalefactor increment is denoted as *sfd*. In Eq. (22), q_o is obtained by re-quantizing x_{Ri} that is the reconstructed MDCT coefficients. Eq. (22) shows that re-quantization can simply be done by applying a scalefactor increment to the original quantized coefficients without the inverse quantization.

$$\begin{aligned} q_o &= \text{int}\left(x_{R_i} \cdot 2^{-\frac{3}{16}\cdot sf_o} + 0.4054 \right) \\ &= \text{int}\left(q_i \cdot 2^{-\frac{3}{16}\cdot (sf_o - sf_i)} + 0.4054 \right) \\ &= \text{int}\left(q_i \cdot 2^{-\frac{3}{16}\cdot sfd} + 0.4054 \right) \end{aligned} \tag{22}$$

With the scalefactor increment, the SNR values for audio signals at the transcoded bitstream are analyzed by Eq.(23). The suffixes R_i and R_o mean the reconstructed signals of input and output bitstreams respectively.

$$
\begin{aligned}
SNR &= 10 * \log\left(\frac{\left|mdct\_line\right|_{R_i}}{\left|mdct\_line\right|_{R_i} - \left|mdct\_line\right|_{R_o}}\right)^2 \\
&= 10 * \log\left(\frac{x_{R_i}^{\frac{4}{3}}}{x_{R_i}^{\frac{4}{3}} - x_{R_o}^{\frac{4}{3}}}\right)^2 \\
&= 10 * \log\left(\frac{q_i^{\frac{4}{3}} \cdot 2^{\frac{1}{4}\cdot sf_i}}{q_i^{\frac{4}{3}} \cdot 2^{\frac{1}{4}\cdot sf_i} - q_o^{\frac{4}{3}} \cdot 2^{\frac{1}{4}\cdot sf_o}}\right)^2 \\
&= 10 * \log\left(\frac{\left(q_i \cdot 2^{-\frac{3}{16}\cdot sfd}\right)^{\frac{4}{3}}}{\left(q_i \cdot 2^{-\frac{3}{16}\cdot sfd}\right)^{\frac{4}{3}} - q_o^{\frac{4}{3}}}\right)^2
\end{aligned}
\tag{23}
$$

By substituting the variables at Eq.(20) into Eq.(22), Eq. (23) shows that the SNR value of the output signals can be derived by a function of the quantized input coefficients q_i and the scalefactor increment *sfd*. Given an input coefficient q_i, we can make an observation on the correlation between SNR and *sfd* according to Eq.(23). Figure 3-9 demonstrates that in most of the cases, the SNR values may decrease as the magnitudes of *sfd* increases.

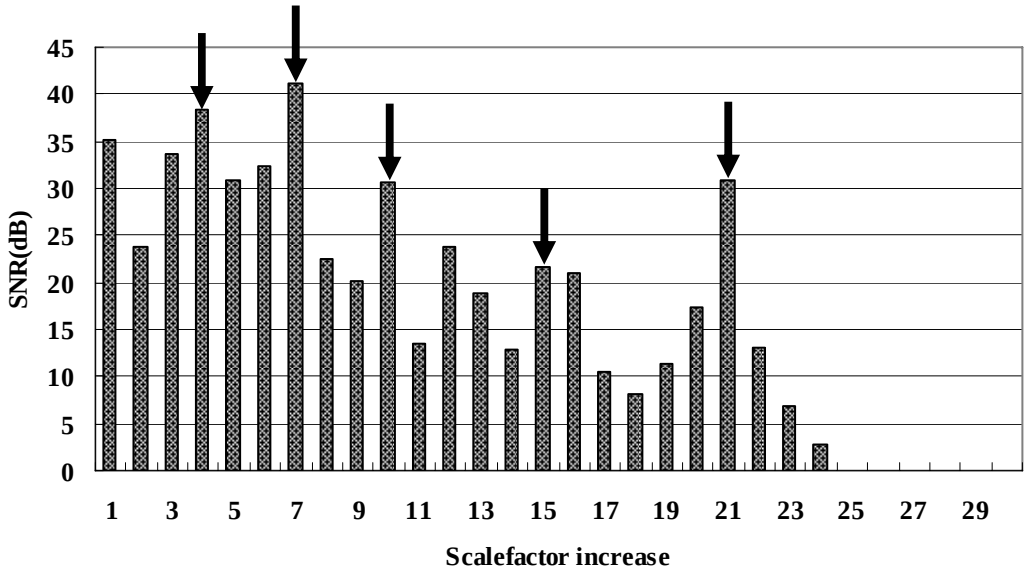

Figure 3-9. SNR value with increasing *sfd* and a constant q_i

3.2.3 NSR-based Rate-Distortion Optimization

The NSR-based rate-distortion optimization is based on NSR-based search for the best scalefactor increment. The NSR measure is derived from SNR. To be consistent with the basic principal of NMR that is used to present the distortion energy over the perceptual masking thresholds, the derivation is defined in Eq. (23) with an offset to make positive NSR values.

$$NSR \equiv round\left(10 * \log 10\left(\frac{\left(q_i \cdot 2^{-\frac{3}{16}\cdot sfd}\right)^{\frac{4}{3}} - q_o^{\frac{4}{3}}}{\left(q_i \cdot 2^{-\frac{3}{16}\cdot sfd}\right)^{\frac{4}{3}}}\right)^2 + 80\right), \tag{24}$$

With Eq. (24), FRDOT can make a connection with the NSR measure and the transcoding process. Given a quantized coefficient q_i from the input bitstream, the NSR value of re-quantized coefficient at the output bitstream q_o can be computed by applying the scalefactor increment *sfd*.

Table 3-1. NSR with increasing *sfd* (q_i=10)

sfd	1	2	3	4	5	6	7	8	9	10	11	12
q_o	9	8	7	6	5	4	4	3	3	3	2	2
NSR	50	54	53	42	55	64	39	66	53	63	67	56

sfd	13	14	15	16	17	18	19	20	21	22	23	24
q_o	2	2	1	1	1	1	1	1	1	0	0	0
NSR	61	70	71	68	61	54	68	74	78	80	80	80

As shown in Table 3-1, the NSR value is increased and q_o is decreased as the scalefactor increment *sfd* is increased from the unity to 24. As mentioned in Section 3.2.1, for varying scalefactor increments, the quantization distortion of a quantized input coefficient may be different. Therefore, some distortion levels might produce a smaller NSR with a larger *sfd*. With a specified range of *sfd*, the minimum NSR is found at the intersection of the dotted line and the curve in Figure 3-5. As illustrated on, when the q_o is quantized to the unity with the *sfd* ranging from 15 to 21, the locally minimal NSR occurs at *sfd* equal to 18 (*NSR* = 54).

Notice that NSR value is saturated to 80 as the coefficient is re-quantized to zero. Thus the optimization based on NMR can be represented by the optimization based on SNR or NSR as shown in Eq. (19).

Based on NSR, we can get the best scalefactor increment for each scalefactor band in the transcoder by exhaustively searching for sfd_{best} from all legal values defined in MPEG-2/4 AAC standard. To search minimal NSR values within the range covering all legal scalefactors is time consuming, which will not fit the real-time requirement. Therefore, the remaining issue is the complexity reduction in finding the minimal NSR values. The complexity to search for the minimal NSR values can be reduced by defining a proper search range. Thus, to reduce the overall computation cost of FRDOT, we present a fast NSR-based search algorithm to get the best scalefactor increment at a given bitrate.

To trade off listening quality for fast transcoding, the NSR-based search algorithm reduces the overall computation complexity based on the starting point prediction, the reduced search range and the rapid quality comparison for varying *sfd*. The starting point is predicted based on the bitrates of the original bitstream and the converted bitstream. The search range is reduced based on the distribution of NSR values that are obtained by simulating AAC transcoding at varying bitrates. The re-quantization and quality comparison are done by a table lookup technique.

Figure 3-10 illustrates the method to decide the search range. The sfd_{est} is estimated with the original and target bitrates. In addition, a forward search range and backward search range are decided. An asymmetric search range is applied with the forward search range larger than the backward search range, which may help in finding the sfd_{best} within the forward search range. When the sfd_{best} is larger, the more coding bits in the scalefactor band are reduced. To save the bits needed for a large scalefactor increment, the asymmetric search range is used to compensate the burden of side information increase.

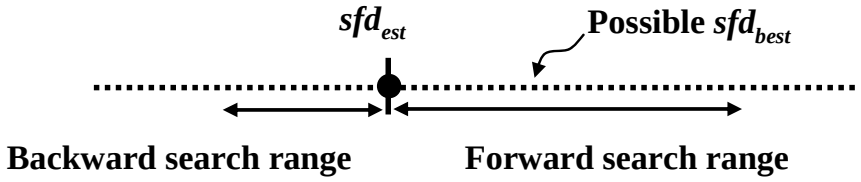

Figure 3-10. Search range of scalefactor increment sfd_{best}

In addition, a proper size of search range is decided according to the distribution of local

SNR maximum, to cover the best reconstruction point described in Section 3.2.1 with an estimated level of the quantized coefficients.

After defining the search range of scalefactor increment sfd_{best}, the requantization is done with a NSR lookup table. The NSR optimized algorithm allocates the bits to each scalefactor band at a given bitrate. From Eq. (6) to Eq. (10), the sfd_{est} is derived as a proper quantization step-size that fits the allocated bits. We can make an assumption that the neighboring *sfd* values of sfd_{est} will fit the bit allocation requirement. The neighbors of sfd_{est} form a range to search for a better *sfd* that lead to a less NSR. For each non-zero quantized coefficients q_i in a certain scalefactor band, we can look for the specified *sfd* with a corresponding NSR on the table. The NSR table is constructed based on Eq. (24). The *NSR* values of non-zero quantized coefficients are summed up in the scalefactor bands. To get the locally minimal NSR value, we look for a sfd_{best} within the predefined search range.

Figure 3-11 depicts a re-quantization process. In the current scalefactor band, we assume that there are four non-zero quantized coefficients q_i , q_i= 1, 1, 4, 2. With the coefficients, a sfd_{est} that fits the bit allocation requirement is calculated. With the sfd_{est}, we search for the sfd_{best} within a search range with *sfd* of the magnitudes in a range of 3 to10. By indexing q_i in the scalefactor band, the corresponding NSR values are found by a table lookup technique. The NSR table is used for looking for the NSR values based on the current q_i. All found NSR values are summed up with *sfd* equal to 3. The summation of NSR *values*, 77, 77, 63 and 70 is 287. After applying the *sfd* from 3 to 10 to the processing scalefactor band, we can find the best candidate of *sfd* that has the minimal NSR summation.

As *sfd* is 5, the NSR summation is 178, which is the minimum within the specified search range. Thus, the best scalefactor increment sfd_{best} is 5 and is added to the scalefactor of the input bitstream to get the final scalefactor. The final scalefactors of every frame are used for the lossless coding module to encode and compute the required bits. The bitrate reduction by NSR-based search algorithm is started from the highest scalefactor band and terminated when the target bitrate of the processing frame is achieved. To meet with the specific bitrate with the smallest NSR value, the table lookup techniques can reduce the processing time. The remaining issue is the size reduction of the lookup tables.

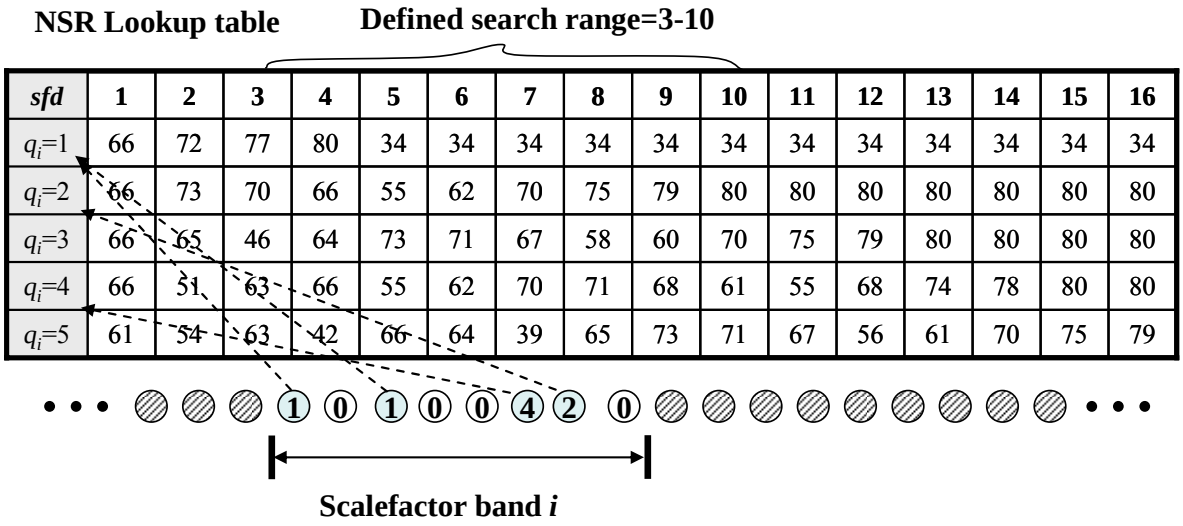

Figure 3-11. An illustration of the re-quantization process

The size reduction of the lookup tables is done based on statistical distribution of quantized input coefficients. Since most of the quantized MDCT coefficients at AAC compressed bitstreams are small even at high bitrates. To investigate the magnitude histogram of the quantized coefficients, an open source code faac [17] and the two test sequences from EBU [21] are used. Figure 3-12 to Figure 3-15 showed that high portions of the coefficients are quantized to small values at 128 kbps and 160 kbps. In addition, more than 98% of quantized coefficients are smaller than 5 in magnitudes.

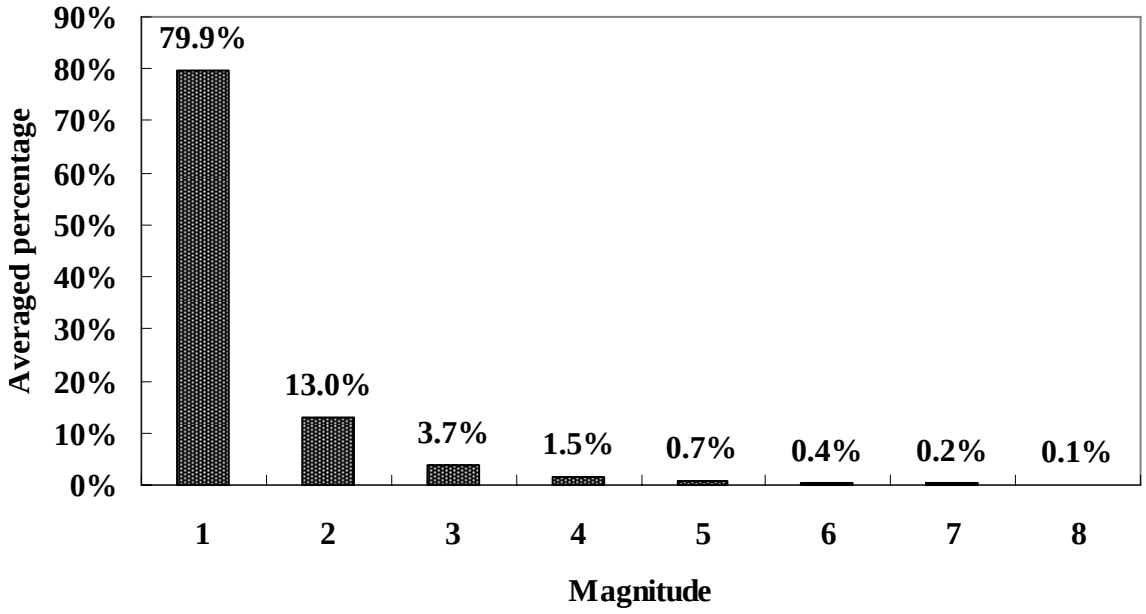

Figure 3-12. Percentage of non-zero quantized coefficients at 128kbps (EBU NO.66)

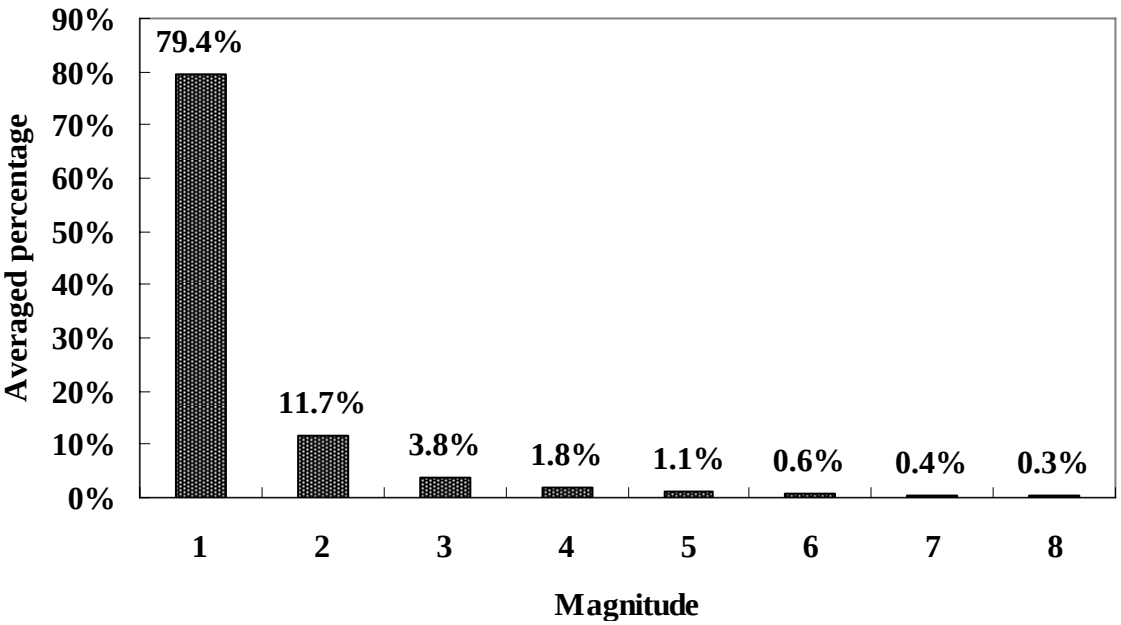

Figure 3-13. Percentage of non-zero quantized coefficients at 160kbps (EBU NO.66)

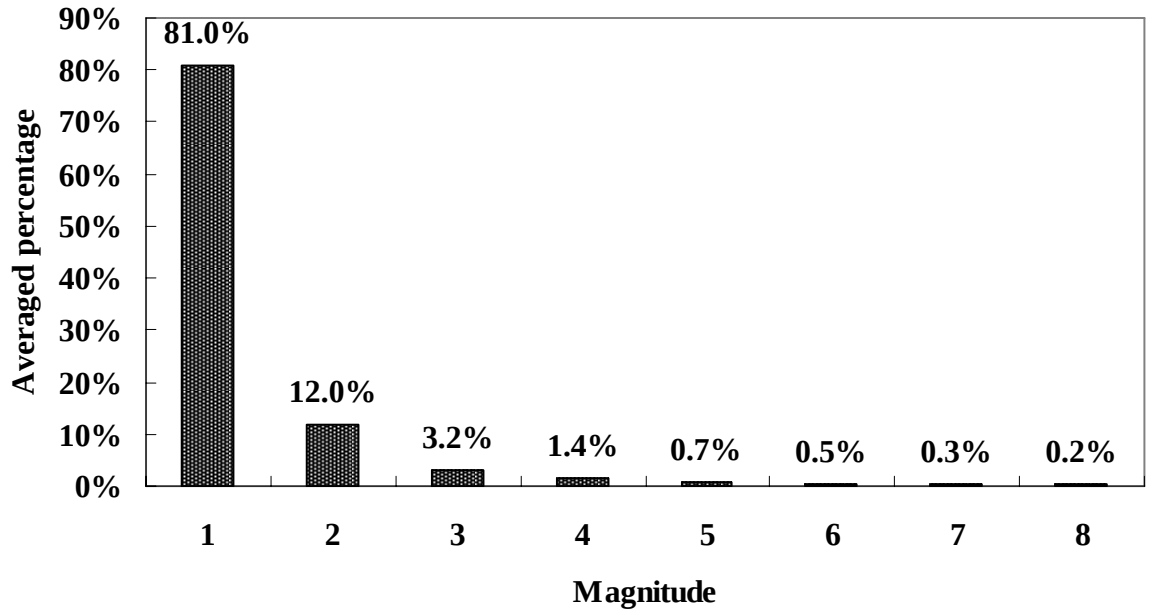

Figure 3-14. Percentage of non-zero quantized coefficients at 128kbps (EBU NO.69)

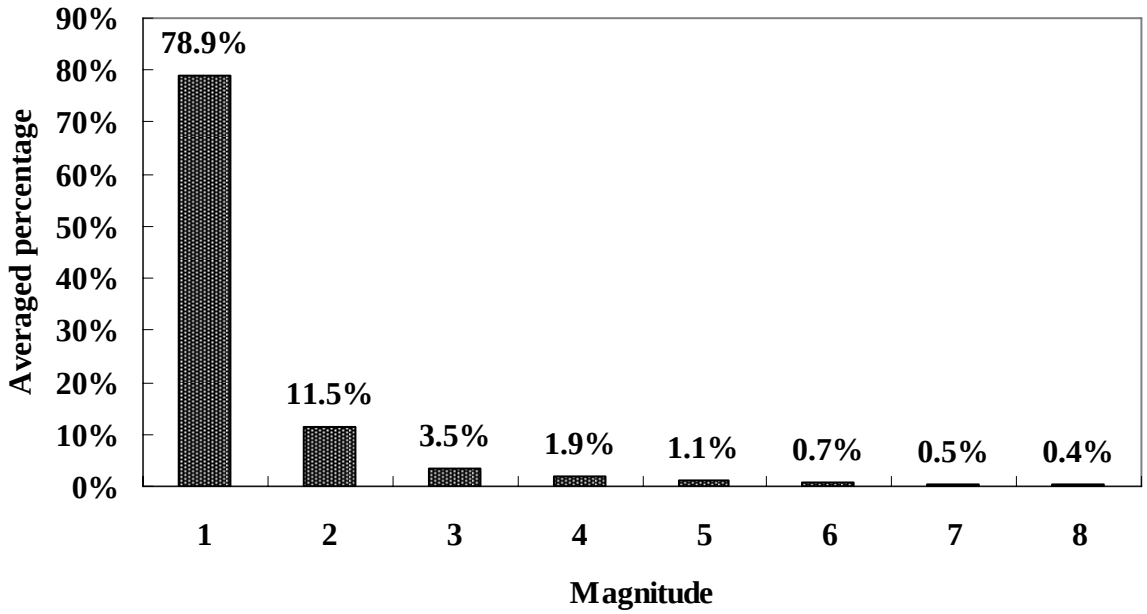

Figure 3-15. Percentage of non-zero quantized coefficients at 160kbps (EBU NO.69)

Our observations showed that more than 98% of non-zero MDCT coefficients are quantized to a level smaller than 5 in magnitudes at high bitrates. Therefore, the index of q_i is only constructed from 1 to 5 to save the memory requirement of the NSR look-up table.

As the target bitrate is much lower than the demanding bitrate, the total number of bits to reduce is enlarged for the coding frame. Thus, the more quantized coefficients will be zeroed out by re-quantization. The basic idea of our algorithm is to make the averaged value of quantized coefficients larger than the unity.

In the NSR table in Figure 3-11, the first row (q_i=1) has a diminished NSR value as the magnitude of *sfd* is larger than 5, which makes the re-quantizing process more flexible because coefficients equal to the unity are allowed to be quantized to zero. In addition, the larger sfd_{est} will be estimated when there are large coefficients in the scalefactor band. Thus, the neighboring unit quantized coefficients (q_i = 1) can be zeroed out by re-quantization because there are stronger masking effects caused by the coefficients of large amplitudes.

3.2.3.1 Multiple Search and Single Search

Figure 3-16 demonstrated the NMR optimized transcoding architecture referring to the architecture of modified SLAT. The bits allocated to each scalefactor band should be larger

than the amount of bits to represent the quantized coefficients with the averaged magnitude equal to the unity. In addition, the re-quantizing process with the NSR table is applied from the highest scalefactor band to lower scalefactor band to rapidly compare the audible quality by varying scalefactor increments. After the best scalefactor increment *sfd* is obtained within the specified search range, the scalefactor increment is added to sfd_{best} and the quantized coefficients are updated by Eq. (22).

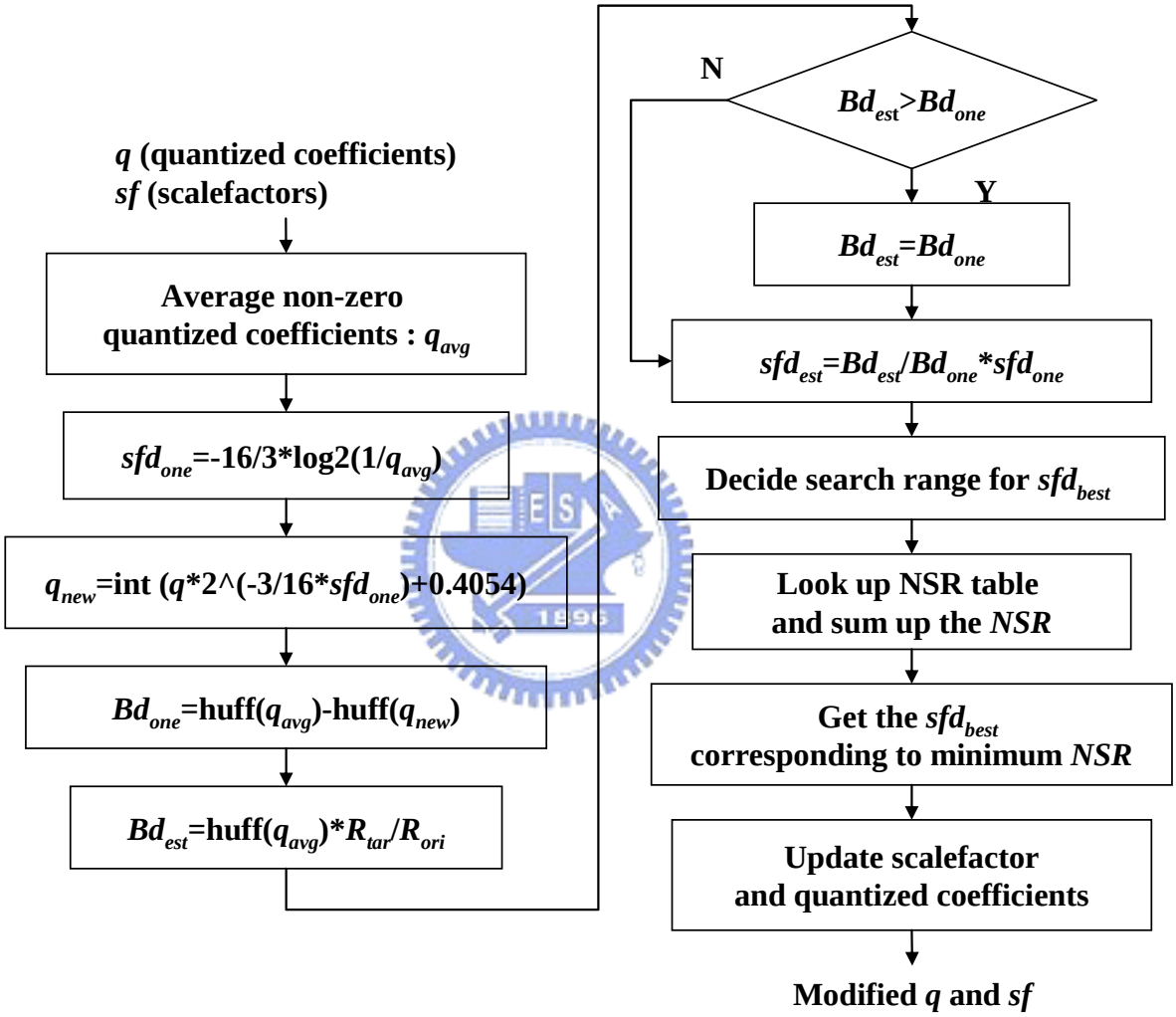

Figure 3-16. Flow chart of NMR optimized transcoding algorithm

The NSR table lookup RDO algorithm is applied to the scalefactor bands from high frequency to low frequency. With the estimate bit budget for each scalefactor band, the remaining number of bits may not meet the bitrate requirement after the RDO process goes through the whole frame by once. There are two methods for the transcoding process to meet the bitrate requirement. The first method is called as multiple search, which repeats the RDO

process from the highest scalefactor band when the RDO process has reached the lowest scalefactor band. The NSR lookup RDO iterates in the current frame until the bitrate requirement is met. The alternative method is called as single search. When the NSR lookup RDO goes through the whole frame by once, a bandwidth limiter is used to ensure that the remaining number of bits meets the bitrate requirement.

Figure 3-17 shows the algorithm flow of rate-distortion optimized transcoding (RDOT) that uses the multiple search method. First, we can obtain quantized coefficients and scalefactors after the noiseless decoding as the same with SLAT. The BCM estimates the number of bits needed to be reduced in the current frame and dynamically adjusts the empirical ratio *BFAC* to make the bitrate reduction more precise. Then the NMR optimized algorithm from high to low frequency bands iterates until the bitrate requirement is met. Finally, the quantized coefficients and scalefactors are updated by Eq. (22) and encoded by the noiseless coding.

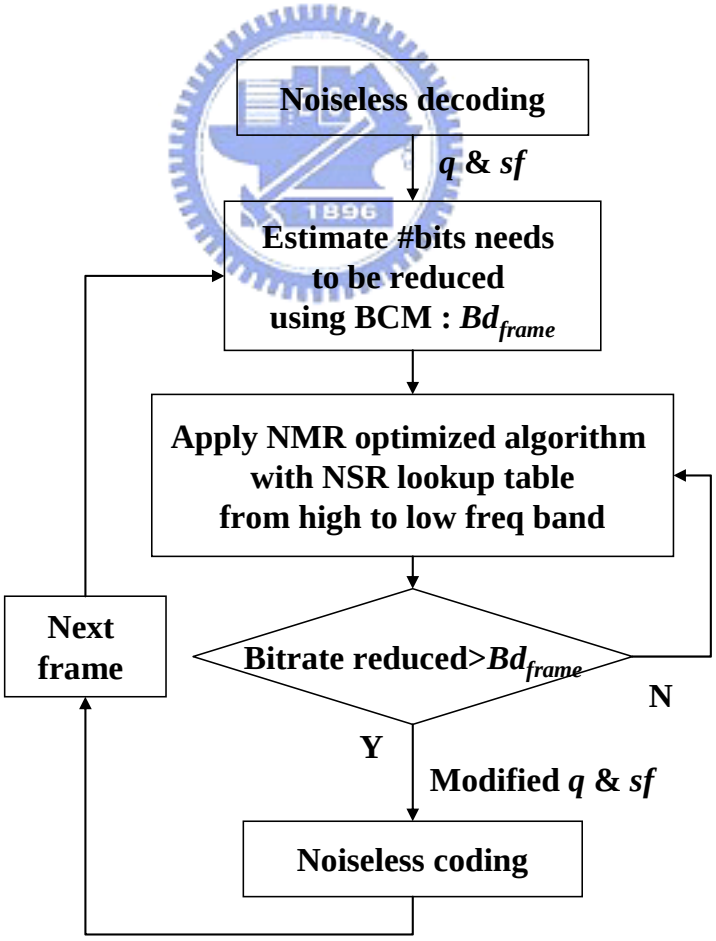

Figure 3-17. Flow chart of RDOT

Figure 3-18 shows the algorithm flow of fast rate-distortion optimized transcoder (FRDOT) which uses the single search method cascaded by the bandwidth limiter. The execution time of FRDOT is less than RDOT by removing the iterative operations within every coding frame.

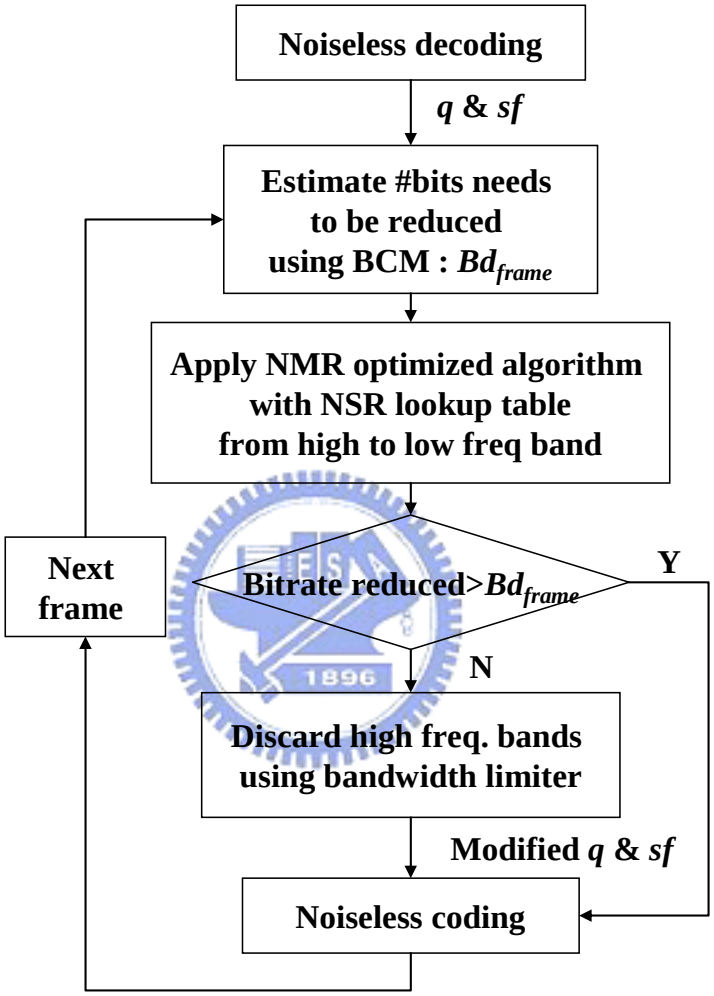

Figure 3-18. Flow chart of FRDOT

RDOT iterates within a frame more times when there are more number of bits to reduce. In addition, the FRDOT spends less time than RDOT and still retains a good transcoding quality from the experiment results. Thus, the FRDOT uses the single search method cascaded by the bandwidth limiter.

3.3 Observations on AAC Short Window Coding

When the block type is the eight short window, the grouping and interleaving

mechanisms are turned on for AAC encoding. In order to improve the coding efficiency, the coefficients associated with neighboring short windows can be grouped to own a set of scalefactors. The coefficients within a group are interleaved by reshuffling the scalefactor bands from each group of short windows. With the grouping and interleaving mechanisms, the side information covering the scalefactors, Huffman code books… etc is reduced.

To investigate the performance of the MPEG-2/4 AAC short window grouping at the target bitrates from 64 kbps to 192 kbps, three different window grouping approaches are used. The 8 test sequences are adopted for simulations and the percentage of the short window blocks are listed on Table 3-2. On Table 3-3, the three grouping methods that combine the 8 short window sequences into 1, 2 and 4 groups are used for performance comparison. The audio quality is measured with NMR and Objective Difference Grade (ODG) [22], which quantifies the degradation of an input signal over the input signal itself. The ODG value is ranged from 0 to -4. As the ODG value decreases, the audible quality of the input signal decreases as well.

The observations showed that the number of groups increases, the coding performance decreases at varying bitrates. Table 3-3 and Table 3-3 show that as the number of groups increases, the percentage of the total bits taken by the side information increases. Thus, the decrease of the coding efficiency is caused by consuming the bits to transfer the side information of grouping at the bitstream with a constant bitrate. For a given bitrate, the coding of side information will decrease the total number of bits to encode the audio content.

Table 3-2. Percentage of the short window blocks within the test sequences

Percentage	**TS01**	**TS02**	**TS03**	**TS04**	**TS05**	**TS06**	**TS07**	**TS08**
	14.45	1.06	28.57	4.31	3.67	20.85	73.14	34.98

Table 3-3. Percentage of side information per frame by applying three different grouping methods (96 kbps)

Percentage	**TS01**	**TS02**	**TS03**	**TS04**	**TS05**	**TS06**	**TS07**	**TS08**
1 group	10.94	14.49	11.96	12.00	10.21	11.18	11.57	11.74
2 groups	17.96	23.74	20.82	20.64	18.44	19.01	19.45	20.46
4 groups	29.83	39.04	33.47	34.71	31.46	31.10	30.86	32.84

Table 3-4. Percentage of side information per frame by applying three different grouping methods (128 kbps)

Percentage	TS01	TS02	TS03	TS04	TS05	TS06	TS07	TS08
1 group	9.79	11.19	9.46	10.21	8.79	9.40	9.72	9.88
2 groups	17.60	21.22	18.45	18.82	15.51	17.15	18.38	18.32
4 groups	27.29	35.11	30.82	31.21	27.41	28.51	29.06	30.55

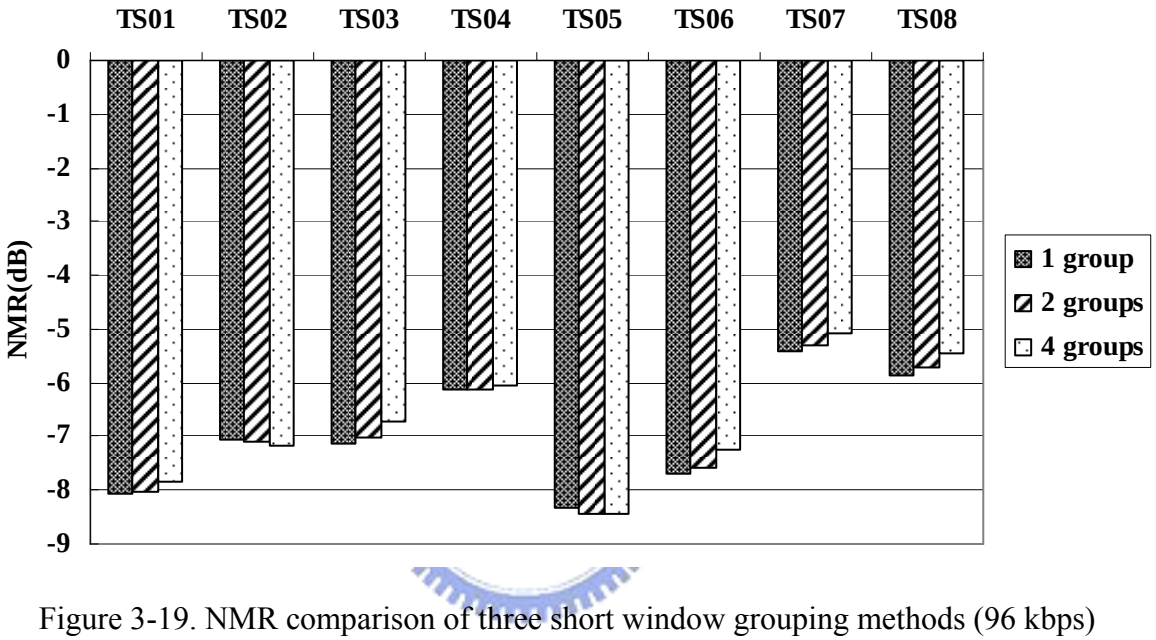

Figure 3-19. NMR comparison of three short window grouping methods (96 kbps)

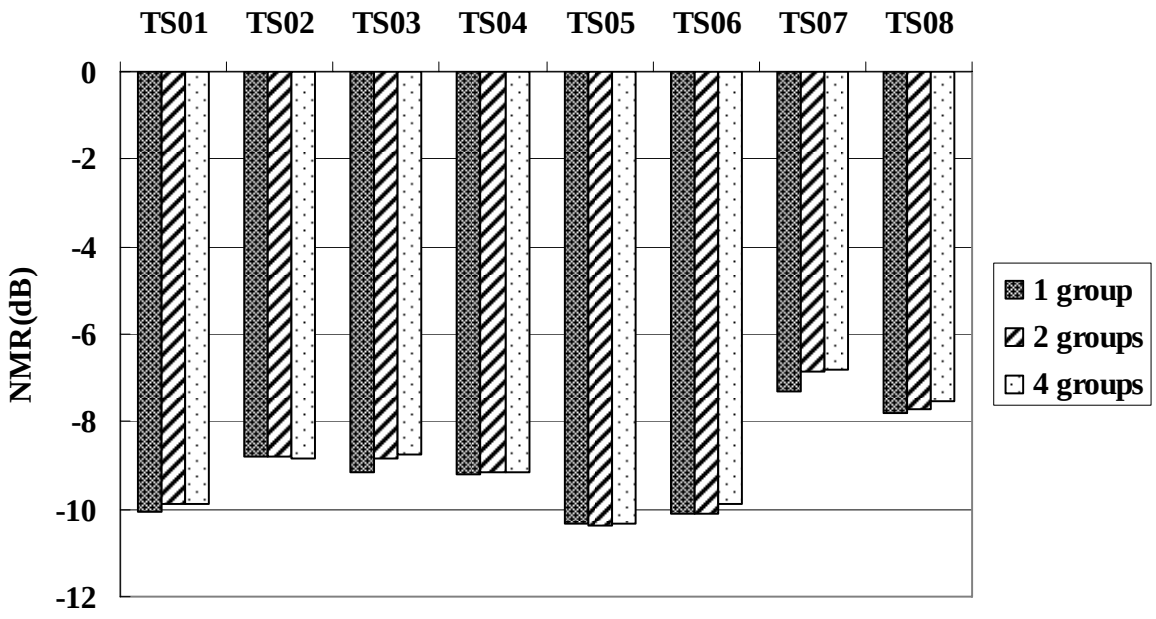

Figure 3-20. NMR comparison of three short window grouping methods (128 kbps)

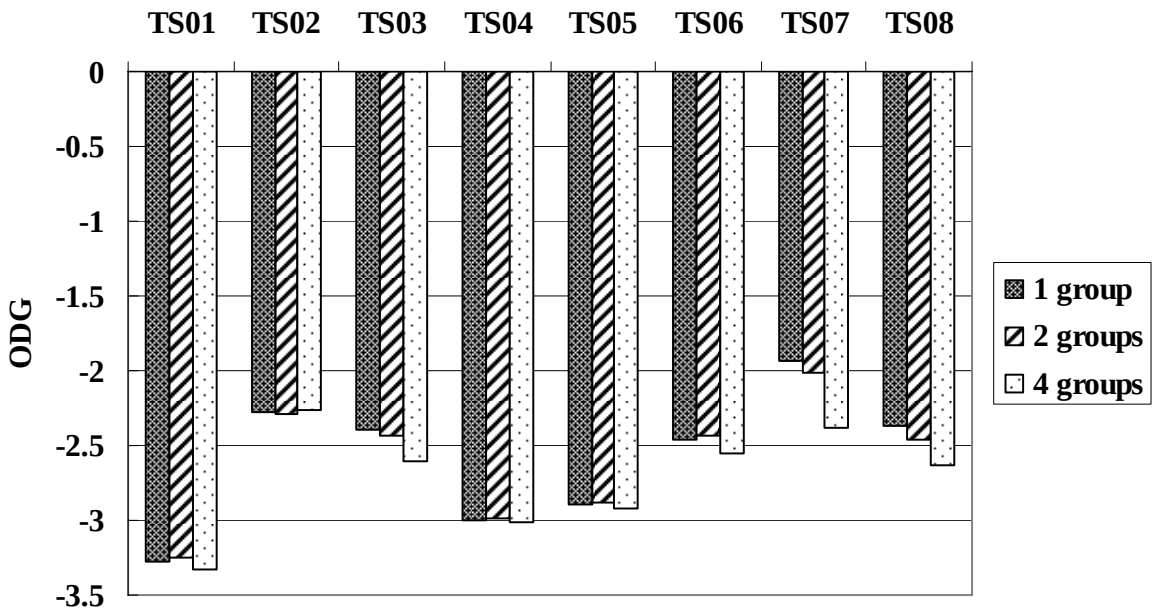

Figure 3-21. ODG comparison of three short window grouping methods (96 kbps)

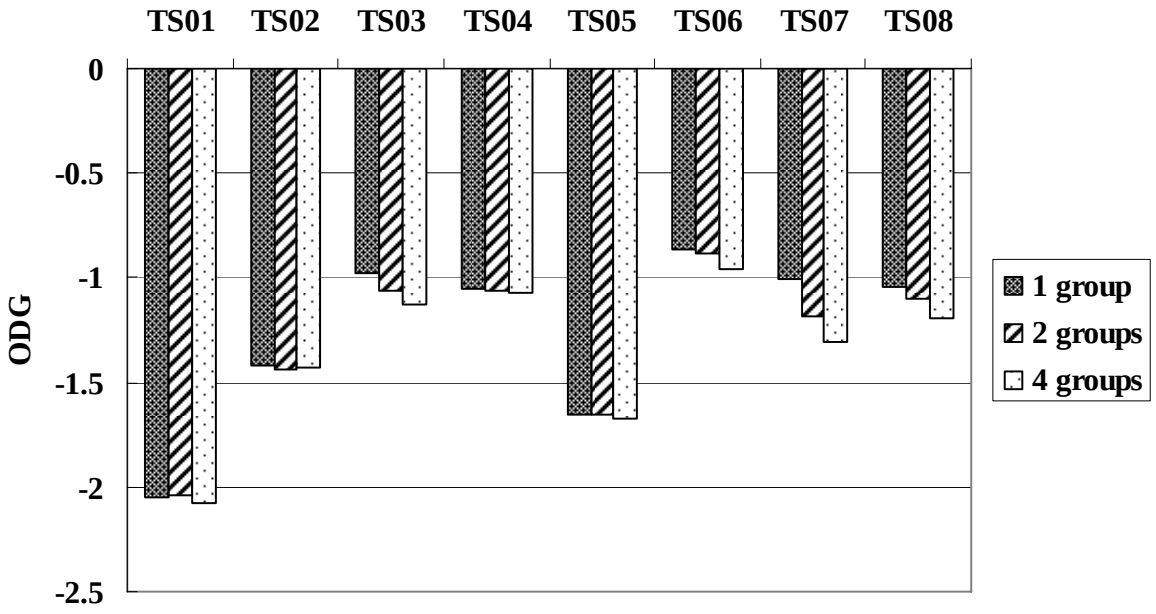

Figure 3-22. ODG comparison of three short window grouping methods (128 kbps)

As shown from Figure 3-19 to Figure 3-22, when a short window frame is split to more groups, the quality will be degraded for most of the test sequences. In addition, for the test sequences having high percentage of short window blocks, the overall quality is a little worse. Thus, for each test sequence, the high percentage of short window blocks and the large number of groups will increase side information overload and lower the coding quality. Consequently, considering the performance of MPEG-2/4 AAC short window coding at the

bitrates, we merged the eight short window sequences into a unitary frame during FRDOT transcoding.

When the single group method is applied, the eight short window sequences share the same set of side information. After grouping and interleaving, there are eight times of MDCT coefficients within each scalefactor band to be quantized with a single scalefactor. To speed up the FRDOT, the ρ-domain model that removes the high frequency coefficients is applied when the transcoding block has eight short windows.

Chapter 4

Experimental Results

To evaluate the rate-distortion performance of RDOT and FRDOT, the experiment environment and the performance measures are introduced. The performance comparison between the proposed algorithms and the other transcoders is based on the rate control efficiency, the audio quality of transcoded bitstream and the execution time.

4.1 Environment

4.1.1 Experiment Parameters

The transcoder consists of a decoder and an encoder. To evaluate the coding performance and execution time, MPEG-2/4 AAC LC based platform is adopted. The source files of the platform are Free Advanced Audio Decoder and Free Advanced Audio Encoder, which are downloaded from [17].

The eight stereo test sequences from European Broadcasting Union (EBU) [21] are encoded at bitrates 128 kbps and 160 kbps for the transcoding sources. In addition, the sampling rate is 44.1 kHz.

Table 4-1. List of test sequences

Test Sequence	Track No.	Content	Suggested Application
TS01	40	Harpsichord	N/A
TS02	12	Piccolo	N/A
TS03	58	Guitar	Aliasing distortion/Overload after processing/Programmed-modulated noise
TS04	48	Quartet	N/A
TS05	42	Accordion	N/A
TS06	66	Wind ensemble	Aliasing distortion/Bitrate reduction/Frequency response
TS07	70	Eddie Rabbitt	Overload after processing/Programmed-modulated noise
TS08	69	ABBA	Bitrate reduction/Stereophonic image

For performance comparison, three bitrate transcoding methods including CT, FRDOT and RDOT are used. In addition, the rate-distortion performance of FAAC is used as the reference.

A. "FAAC" data are encoded directly with the FAAC encoder at different bitrate.
B. "CT" represents the data transcoded by the cascaded transcoder.
C. "FRDOT" data are generated by the proposed algorithm with single search.
D. "RDOT" data are generated by the proposed algorithm with multiple search.

4.1.2 Performance Measures

The listening quality of the transcoded bitstream and the execution time of the transcoders are used for comparison. The quality is measured by NMR that is the ratio of noise-to-masking in decibel. The NMR of zero value means the imperceptible quality. The ODG (Objective Difference Grade) [22] that shows the difference grade to show the impairments is also used for performance comparison. Both NMR and ODG are calculated by EAQUAL software [23]. The improvement on the execution time is measured by profiling in Microsoft Visual C++ 6.0. Thus, the speedup can be defined as the execution time of CT

divided by the execution time of FRDOT.

Table 4-2. Objective Difference Grade

Difference Grade	Description of Impairments
0	Imperceptible
-1	Perceptible but not annoying
-2	Slightly annoying
-3	Annoying
-4	Very annoying

4.2 Results and Remarks

The transcoders including FRDOT, CT and FAAC were compared. Table 4-3 shows the averaged bitrate of transcoding output. The transcoding output is for the four different methods when the transcoding source is archived at 128kbps. The results showed that the averaged bitrate of FRDOT is closer to the target bitrate than CT and FAAC.

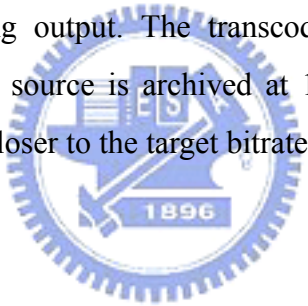

Table 4-3. Averaged bitrates of FRDOT, CT and FAAC

(a) FRDOT

Bitrate (kbps)	TS01	TS02	TS03	TS04	TS05	TS06	TS07	TS08	Average
64	65.34	64.70	65.49	65.74	64.45	65.27	65.78	64.52	65.16
80	81.02	79.59	80.18	80.79	80.27	80.55	81.29	79.59	80.41
96	96.02	95.66	95.67	96.35	94.92	95.84	96.26	94.94	95.71
112	111.02	110.99	110.49	110.43	109.38	110.54	111.24	110.00	110.51

(b) CT

Bitrate (kbps)	TS01	TS02	TS03	TS04	TS05	TS06	TS07	TS08	Average
64	62.23	64.77	65.42	66.70	64.72	65.27	64.18	65.38	64.83
80	75.28	75.94	80.85	81.53	80.18	81.14	79.68	80.72	79.42
96	92.71	86.16	95.67	96.35	94.92	95.84	95.73	95.22	94.07
112	109.07	108.80	111.17	111.91	111.19	111.72	112.31	111.42	110.95

(c) FAAC

Bitrate (kbps)	TS01	TS02	TS03	TS04	TS05	TS06	TS07	TS08	Average
64	64.56	65.57	66.03	67.15	65.72	65.85	66.31	65.94	65.89
80	80.34	81.05	81.52	82.27	81.45	81.73	81.82	81.58	81.47
96	96.51	96.39	97.02	97.83	96.72	97.02	97.33	97.21	97.00
112	111.99	112.45	112.51	113.40	112.09	112.30	113.38	112.56	112.59

4.2.1 Quality Comparison

The testing samples are encoded to the bitstreams with the bitrates of 128 kbps and 160 kbps. For quality comparison at varying bitrates, the NMR values of different transcoders are shown on Table 4-4 and Table 4-5. The averaged NMR values of eight sequences with different input bitrates are shown in Figure 4-1 and Figure 4-2.

Table 4-4. NMR values with the input sequences at 128 kbps

(a) FRDOT

NMR(dB)	TS01	TS02	TS03	TS04	TS05	TS06	TS07	TS08	Avg
112 kbps	-8.58	-7.11	-7.61	-6.95	-8.64	-8.60	-6.90	-6.67	-7.63
96 kbps	-7.64	-6.69	-6.39	-6.20	-7.95	-7.47	-6.02	-5.80	-6.77
80 kbps	-6.44	-5.44	-5.23	-5.01	-6.79	-6.42	-4.75	-4.71	-5.60
64 kbps	-3.67	-3.34	-3.75	-3.35	-3.87	-5.15	-3.32	-3.37	-3.73

(b) RDOT

NMR(dB)	TS01	TS02	TS03	TS04	TS05	TS06	TS07	TS08	Avg
112 kbps	-8.57	-7.12	-7.61	-6.95	-8.64	-8.601	-6.90	-6.66	-7.63
96 kbps	-7.66	-6.68	-6.40	-6.19	-7.93	-7.474	-6.03	-5.80	-6.77
80 kbps	-6.44	-5.45	-5.24	-5.07	-6.75	-6.457	-4.79	-4.75	-5.62
64 kbps	-3.74	-3.54	-3.81	-3.67	-3.93	-5.257	-3.38	-3.38	-3.84

(c) CT

NMR(dB)	TS01	TS02	TS03	TS04	TS05	TS06	TS07	TS08	Avg
112 kbps	-5.34	-3.94	-4.89	-3.76	-4.97	-5.61	-3.55	-3.60	-4.46
96 kbps	-4.97	-4.07	-4.51	-3.66	-4.66	-5.56	-2.99	-3.66	-4.26
80 kbps	-4.54	-3.73	-3.94	-3.12	-3.82	-4.90	-2.62	-3.38	-3.75
64 kbps	-3.92	-3.37	-3.05	-2.32	-2.99	-4.10	-2.21	-2.70	-3.08

(d) FAAC

NMR(dB)	TS01	TS02	TS03	TS04	TS05	TS06	TS07	TS08	Avg
112 kbps	-9.06	-7.75	-8.31	-7.57	-9.39	-9.08	-7.06	-7.48	-8.21
96 kbps	-8.24	-6.90	-7.23	-5.93	-8.28	-7.72	-6.24	-6.39	-7.12
80 kbps	-7.09	-6.05	-5.70	-4.52	-6.26	-6.33	-5.16	-5.11	-5.78
64 kbps	-5.47	-5.23	-4.28	-2.94	-4.31	-5.04	-4.05	-3.85	-4.40

Table 4-5. NMR values with the input sequences at 160 kbps

(a) FRDOT

NMR(dB)	TS01	TS02	TS03	TS04	TS05	TS06	TS07	TS08	Avg
128 kbps	-9.94	-7.85	-7.96	-7.03	-9.80	-8.99	-8.25	-7.17	-8.37
112 kbps	-8.80	-7.39	-6.69	-6.46	-9.28	-7.94	-7.04	-6.30	-7.49
96 kbps	-7.71	-6.78	-5.81	-5.96	-8.11	-7.17	-5.59	-5.46	-6.57
80 kbps	-5.55	-5.49	-4.81	-4.80	-5.16	-6.29	-4.17	-4.38	-5.08
64 kbps	-2.96	-3.29	-3.41	-3.31	-2.36	-5.04	-2.96	-3.17	-3.31

(b) RDOT

NMR(dB)	TS01	TS02	TS03	TS04	TS05	TS06	TS07	TS08	Avg
128 kbps	-9.94	-7.84	-7.96	-7.03	-9.80	-8.99	-8.25	-7.16	-8.37
112 kbps	-8.81	-7.38	-6.69	-6.45	-9.28	-7.92	-7.03	-6.28	-7.48
96 kbps	-7.82	-6.79	-5.81	-5.86	-8.09	-7.15	-5.59	-5.44	-6.57
80 kbps	-6.11	-5.49	-4.86	-4.84	-5.21	-6.24	-4.16	-4.32	-5.15
64 kbps	-4.29	-3.62	-3.53	-3.62	-3.31	-5.07	-2.99	-3.01	-3.68

(c) CT

NMR(dB)	TS01	TS02	TS03	TS04	TS05	TS06	TS07	TS08	Avg
128 kbps	-7.16	-5.84	-5.82	-4.72	-7.32	-6.64	-4.74	-4.69	-5.87
112 kbps	-6.50	-5.34	-5.39	-4.31	-6.72	-6.30	-4.39	-4.68	-5.45
96 kbps	-6.18	-4.86	-4.97	-3.77	-6.19	-5.73	-4.02	-4.27	-5.00
80 kbps	-5.47	-4.47	-4.29	-3.18	-5.14	-5.04	-3.57	-3.65	-4.35
64 kbps	-4.60	-3.91	-3.42	-2.36	-3.69	-4.27	-3.06	-2.98	-3.54

(d) FAAC

NMR(dB)	TS01	TS02	TS03	TS04	TS05	TS06	TS07	TS08	Avg
128 kbps	-10.25	-8.59	-9.25	-8.82	-10.30	-10.15	-8.06	-8.42	-9.23
112 kbps	-9.06	-7.75	-8.31	-7.57	-9.39	-9.08	-7.06	-7.48	-8.21
96 kbps	-8.24	-6.90	-7.23	-5.93	-8.28	-7.72	-6.24	-6.39	-7.12

80 kbps	-7.09	-6.05	-5.70	-4.52	-6.26	-6.33	-5.16	-5.11	-5.78
64 kbps	-5.47	-5.23	-4.28	-2.94	-4.31	-5.04	-4.05	-3.85	-4.40

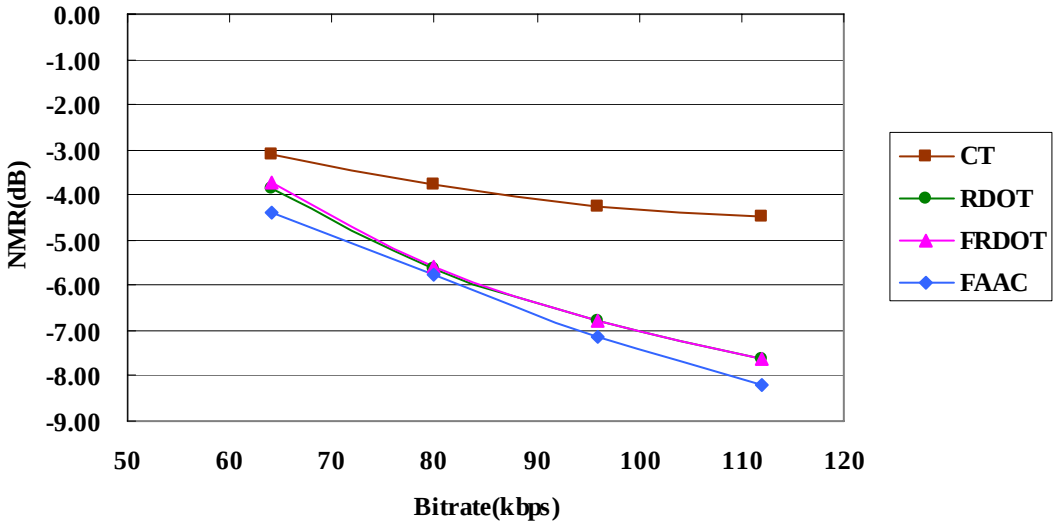

Figure 4-1. Averaged NMR values with the input sequences at 128 kbps

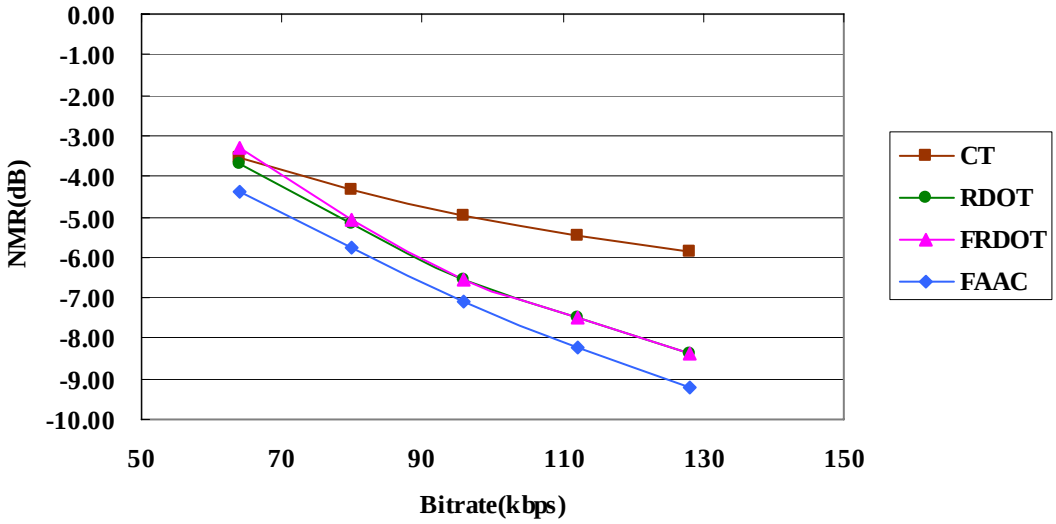

Figure 4-2. Averaged NMR values with the input sequences at 160 kbps

Similarly, the ODG values of eight sequences are shown on Table 4-6 and

Table 4-7. The averaged ODG values are shown in Figure 4-3 and Figure 4-4.

Table 4-6. ODG value with the input sequence at 128 kbps

(a) FRDOT

ODG	TS01	TS02	TS03	TS04	TS05	TS06	TS07	TS08	Avg
112kbps	-3.40	-2.67	-2.21	-3.14	-2.64	-2.09	-1.70	-2.30	-2.52
96kbps	-3.66	-3.03	-3.10	-3.39	-3.00	-3.02	-2.83	-3.14	-3.15
80kbps	-3.75	-3.41	-3.49	-3.62	-3.67	-3.44	-3.44	-3.51	-3.54
64kbps	-3.73	-3.35	-3.59	-3.63	-3.69	-3.59	-3.63	-3.60	-3.60

(b) RDOT

ODG	TS01	TS02	TS03	TS04	TS05	TS06	TS07	TS08	Avg
112kbps	-3.39	-2.61	-2.21	-3.14	-2.66	-2.09	-1.70	-2.31	-2.51
96kbps	-3.65	-3.03	-3.07	-3.37	-2.99	-3.00	-2.79	-3.11	-3.13
80kbps	-3.75	-3.46	-3.48	-3.62	-3.69	-3.38	-3.42	-3.50	-3.54
64kbps	-3.71	-3.57	-3.61	-3.60	-3.68	-3.48	-3.63	-3.62	-3.61

(c) CT

ODG	TS01	TS02	TS03	TS04	TS05	TS06	TS07	TS08	Avg
112kbps	-3.57	-3.36	-2.98	-3.55	-3.24	-3.07	-2.93	-3.08	-3.22
96kbps	-3.73	-3.70	-3.46	-3.68	-3.59	-3.43	-3.40	-3.40	-3.55
80kbps	-3.80	-3.71	-3.65	-3.53	-3.76	-3.66	-3.75	-3.67	-3.69
64kbps	-3.84	-3.74	-3.63	-3.45	-3.78	-3.62	-3.87	-3.73	-3.71

(d) FAAC

ODG	TS01	TS02	TS03	TS04	TS05	TS06	TS07	TS08	Avg
112kbps	-2.69	-1.83	-1.54	-2.21	-2.26	-1.52	-1.31	-1.52	-1.86
96kbps	-3.24	-2.33	-2.37	-3.14	-2.91	-2.45	-1.79	-2.30	-2.57
80kbps	-3.62	-2.95	-3.07	-3.21	-3.51	-3.09	-2.63	-3.09	-3.15

64kbps	-3.80	-3.34	-3.27	-3.25	-3.64	-3.39	-3.52	-3.48	-3.46

Table 4-7. ODG value with the input sequence at 160 kbps

(a) FRDOT

ODG	TS01	TS02	TS03	TS04	TS05	TS06	TS07	TS08	Avg
128kbps	-2.28	-2.33	-1.75	-2.98	-1.84	-1.54	-1.25	-1.87	-1.98
112kbps	-3.20	-2.70	-2.61	-3.12	-2.20	-2.39	-2.20	-2.70	-2.64
96kbps	-3.65	-3.17	-3.26	-3.47	-3.26	-3.12	-3.22	-3.30	-3.31
80kbps	-3.77	-3.29	-3.57	-3.61	-3.66	-3.53	-3.59	-3.58	-3.58
64kbps	-3.69	-3.33	-3.62	-3.60	-3.58	-3.58	-3.67	-3.63	-3.59

(b) RDOT

ODG	TS01	TS02	TS03	TS04	TS05	TS06	TS07	TS08	Avg
128kbps	-2.28	-2.35	-1.75	-2.98	-1.84	-1.53	-1.25	-1.86	-1.98
112kbps	-3.20	-2.70	-2.60	-3.08	-2.21	-2.34	-2.19	-2.61	-2.62
96kbps	-3.64	-3.14	-3.24	-3.42	-3.26	-3.01	-3.19	-3.23	-3.27
80kbps	-3.76	-3.47	-3.56	-3.62	-3.64	-3.45	-3.59	-3.56	-3.58
64kbps	-3.76	-3.61	-3.65	-3.62	-3.60	-3.36	-3.68	-3.62	-3.61

(b) CT

ODG	TS01	TS02	TS03	TS04	TS05	TS06	TS07	TS08	Avg
128kbps	-2.99	-3.12	-2.11	-3.00	-2.32	-2.24	-2.04	-2.29	-2.51
112kbps	-3.38	-3.24	-2.78	-3.32	-2.89	-2.78	-2.48	-2.68	-2.94
96kbps	-3.65	-3.43	-3.29	-3.66	-3.44	-3.32	-3.19	-3.21	-3.40
80kbps	-3.78	-3.60	-3.58	-3.51	-3.69	-3.65	-3.67	-3.60	-3.64
64kbps	-3.83	-3.69	-3.54	-3.45	-3.72	-3.63	-3.82	-3.70	-3.67

(c) FAAC

ODG	TS01	TS02	TS03	TS04	TS05	TS06	TS07	TS08	Avg
128kbps	-2.00	-1.46	-0.96	-1.24	-1.67	-0.85	-0.89	-1.01	-1.26
112kbps	-2.69	-1.83	-1.54	-2.21	-2.26	-1.52	-1.31	-1.52	-1.86
96kbps	-3.24	-2.33	-2.37	-3.14	-2.91	-2.45	-1.79	-2.30	-2.57

80kbps	-3.62	-2.95	-3.07	-3.21	-3.51	-3.09	-2.63	-3.09	-3.15
64kbps	-3.80	-3.34	-3.27	-3.25	-3.64	-3.39	-3.52	-3.48	-3.46

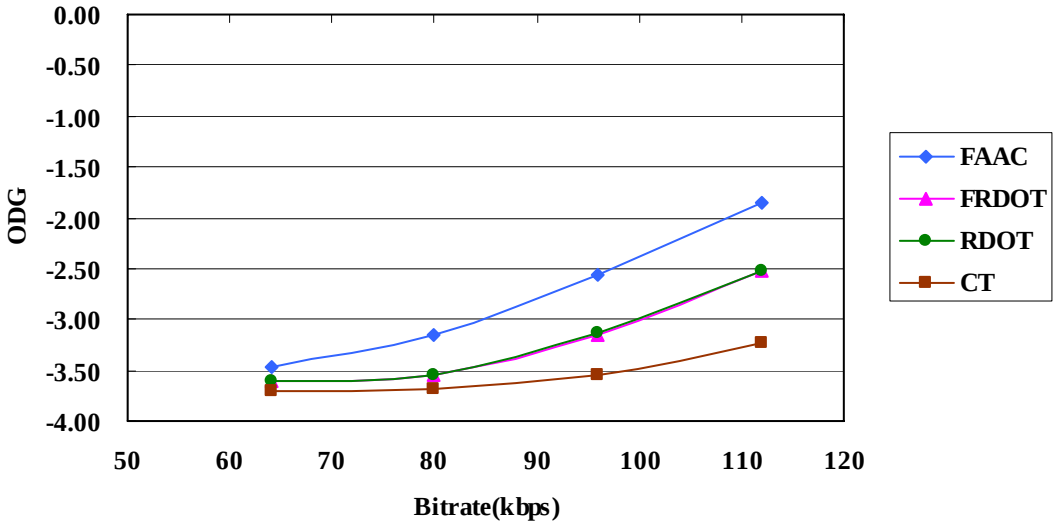

Figure 4-3. Averaged ODG values with the input sequences at 128 kbps

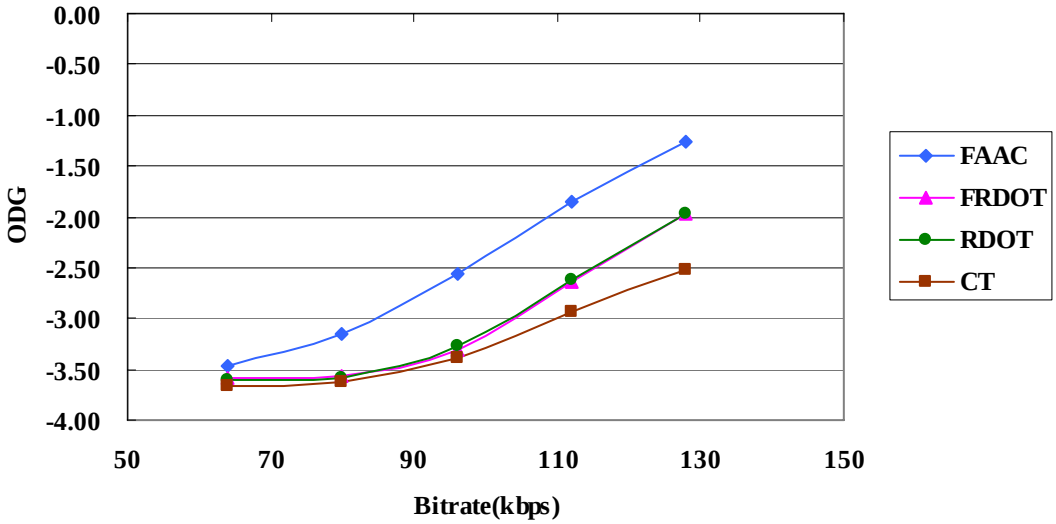

Figure 4-4. Averaged ODG values with the input sequences at 160 kbps

Based NMR and ODG comparisons, FRDOT that uses the bandwidth limiter to avoid the iterative process within every frame can retain a good audible quality. The results show that both the NMR and ODG values of FRDOT are better than those of CT. In fact, CT may be not

the best method for bitrate transcoding. When the signals are completely decoded to the waveforms, the reconstruction error presents. During the encoding, the reconstruction error propagates from PAM to the two nested loop quantizer. As shown in Figure 4-5, the NMR value of original uncompressed source is apparently different from the NMR value of encoded bitstream at high bitrates (MPEG-2/4 AAC at 128kbps), which means the R-D control process will adopt on the error PAM information.

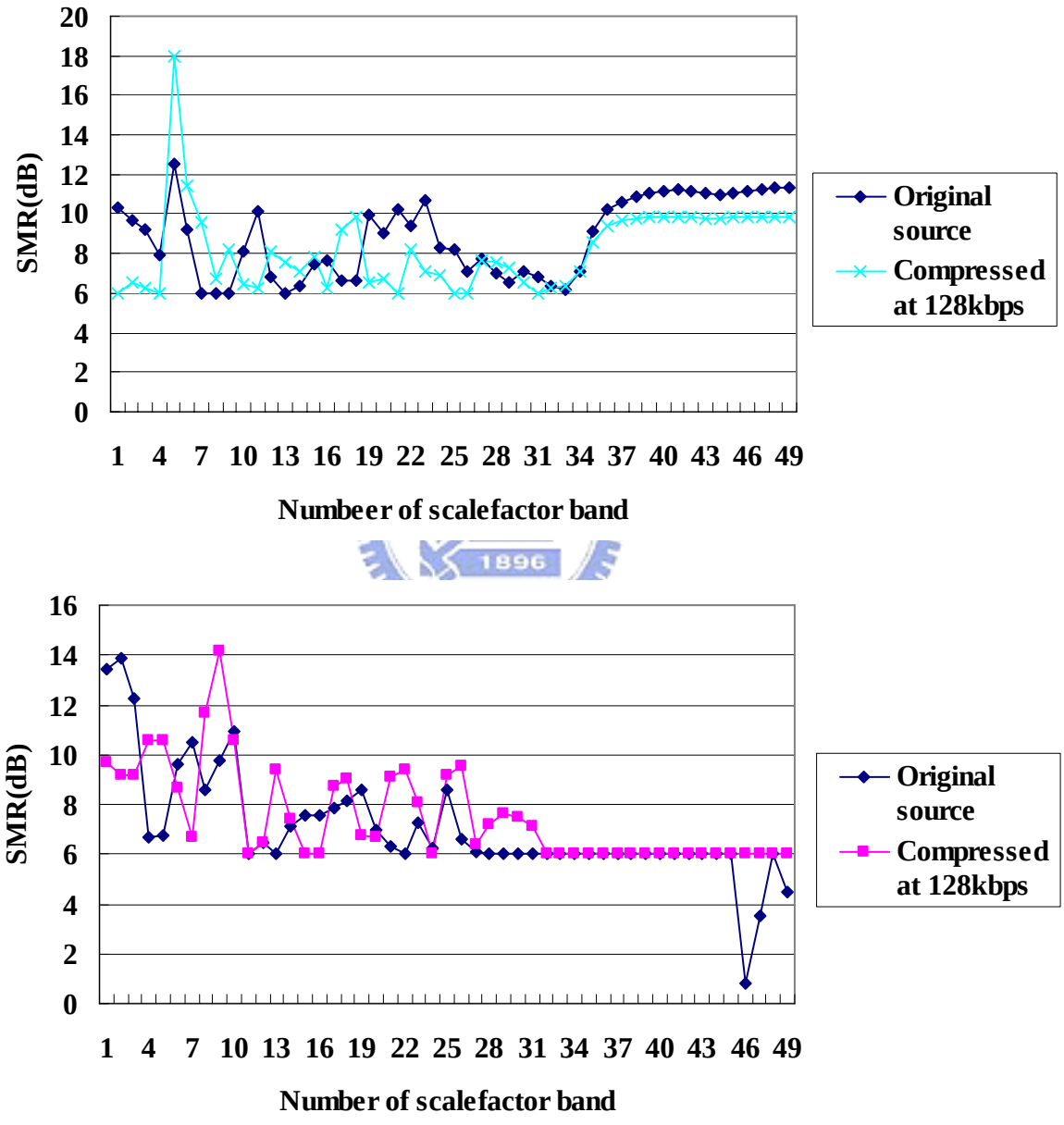

Figure 4-5. PAM information (SMR) comparison of the original source and the compressed bitstreams (128kbps)

The NMR optimized algorithm within FRDOT did not include the PAM module. The re-quantization process is based on the optimization of NSR values in a sense of using the masking information from the original source waveform. The proposed algorithm can perform better than the cascaded transcoding algorithm. When the bitrate of input bitstream increases from 128 kbps to 160 kbps, the NMR value of CT slightly increases. The quality loss by FRDOT is less than 0.5 dB.

4.2.2 Execution Time

Table 4-8. Execution time of FRDOT and CT

(a) Target bitrate at 64 kbps

Execution time (ms)	FRDOT	CT	Speedup
TS01	1506.67	8976.86	5.96
TS02	1876.54	13950.36	7.43
TS03	2056.71	13080.10	6.36
TS04	1927.80	12123.35	6.29
TS05	1728.21	10169.28	5.88
TS06	2336.52	12650.27	5.41
TS07	2366.13	13304.20	5.62
TS08	4485.70	27616.96	6.16

(b) Target bitrate at 80 kbps

Execution time (ms)	FRDOT	CT	Speedup
TS01	1472.85	9527.58	6.47
TS02	1976.04	14820.40	7.50
TS03	2142.74	14064.85	6.56
TS04	2013.75	13358.86	6.63
TS05	1802.01	11096.01	6.16
TS06	2315.53	14162.97	6.12
TS07	2281.76	14189.57	6.22
TS08	4738.01	30991.32	6.54

(c) Target bitrate at 96 kbps

Execution time (ms)	FRDOT	CT	Speedup
TS01	1483.69	10097.50	6.81
TS02	1953.99	15255.90	7.81
TS03	2064.48	15102.33	7.32
TS04	1927.72	14529.39	7.54
TS05	1724.09	11918.49	6.91
TS06	2309.95	15476.44	6.70
TS07	2362.43	15044.64	6.37
TS08	4559.58	32342.60	7.09

(d) Target bitrate at 112 kbps

Execution time (ms)	FRDOT	CT	Speedup
TS01	1455.37	10409.99	7.15
TS02	1866.96	15717.70	8.42
TS03	1997.52	15344.27	7.68
TS04	1848.12	14868.23	8.05
TS05	1633.94	12337.01	7.55
TS06	2260.74	16755.97	7.41
TS07	2422.03	15988.21	6.60
TS08	4548.57	34319.95	7.55

The speedup is defined as the execution time of CT divided by the execution time of FRDOT. The execution time of eight test sequences at different levels of target bitrates is used for comparisons. As shown in Table 4-8, FRDOT is faster than CT by about 5 to 8 times. In addition, the speedup is increased as the target bitrates are enlarged, since the averaged number of bits to reduce is decreased and the processing time of FRDOT is shortened.

Chapter 5

Conclusion

In this chapter, we highlight the innovations of the proposed transcoding algorithm and give some conclusions on FRDOT. In addition, we draw some future work on the possible application of FRDOT to advanced audio coding standards.

5.1 Contributions

In this thesis, we presented a fast bitrate transcoding algorithm called as the FRDOT for real-time audio delivery applications. The major idea of the transcoder is to retain a better quality at a given bit budget under the NMR criterion. The NMR optimized transcoding is based on the search of the best scalefactor increment for the best rate-distortion performance. To measure the distortion without the source audio signals, the NMR measure is represented by the NSR measure. Based on the NSR measure, we present a fast search algorithm of the best scalefactor increment at a given bitrate. To speed up the search, we reduce the total number of scalefactor increments to decrease the computation of NSR values. The reduced search range is found empirically. In addition, the NSR computation is saved by the table lookup technique. The lookup table presents the relationship between the scalefactor increments and the NSR values. The experiment results show that FRDOT is better than CT by 0.5-3 dB in NMR at different target bitrates. In addition, the results show that FRDOT is faster than CT by 5-8 times on the average.

In the FRDOT architecture, BCM provides a model to estimate the bit difference for spectral coefficients between the original and target bitrates. In addition, the BCM makes the averaged bitrate of transcoded bitstream close to the target bitrate. The results show that the averaged output bitrate of FRDOT is closer to the target bitrate than that of CT.

5.2 Future Works

MPEG-4 HE-AAC is the advanced compression approach for audio coding. MPEG-4 HE-AAC is applicable to the applications like satellite-delivered digital audio broadcast and

mobile telephony audio streaming. The audio content delivery applications require the bitrate adaptation mechanism. Since the MPEG-4 HE-AAC consists of MPEG-2/4 AAC system with LC Profile. Thus, the NMR optimized algorithm in FRDOT can be applied to MPEG-4 AAC and HE-AAC. The additional effort is to fit the AAC transcoding techniques with Spectral Band Replication (SBR) and Parametric Coding (PS) modules [5].

References

[1] J. Herre and B. Grill, "Overview of MPEG-4 audio and its applications in mobile communications," in *Proc. WCCC-ICSP*, vol. 1, pp. 11–20, Aug. 2000.

[2] K. Brandenburg, "MP3 and AAC explained," *AES 17th Int. Conf. High-Quality Audio Coding*, Italy, Aug. 1999.

[3] J. Zhou and J. Li, "Scalable audio streaming over the Internet with network-aware rate-distortion optimization," in *Proc. IEEE ICME*, vol. 2, Aug. 2001, pp. 567-570.

[4] S. Quackenbush, "MPEG technologies: advanced audio coding," *ISO/IEC JTC1/SC29/WG11*, Nice, FR, Oct. 2005.

[5] 3GPP TS 26.401, "General audio codec audio processing functions; Enhanced aacPlus general audio codec; General description," Mar. 2005.

[6] ISO, Information technology-Generic Coding of Moving Pictures and Associated Audio, 1997. ISO/IEC JTC1/SC29, ISO/IEC IS 13818-7 (Part 7, Advanced audio coding).

[7] ISO, *Information technology–Coding of Audio-Visual Objects*, 1999. ISO/IEC JTC1/SC29, ISO/IEC IS 14496-3 (Part 3, Audio).

[8] H. Park, et al., "Multi-layer bit-sliced bit rate scalable audio coding," *AES 103rd Convention*, New York, Aug. 1997 (preprint 4520).

[9] H. Hartenstein, et al., "High quality mobile communication," in *Proc. of KIVS 2001*, Hamburg, Feb. 2001.

[10] A. Kassler and A. Schorr, "Generic QoS aware media stream transcoding and adaptation," in *Proc. of Packet Video Workshop*, Nantes, France, Apr. 2003.

[11] http://www.shoutcast.com

[12] http://www.allofmp3.com

[13] Y. Takamizawa, et al., "High-quality and processor-efficient implementation of an MPEG-2 AAC encoder," in *Proc. ICASSP*, vol. 2, May 2001, pp. 985-988.

[14] T. Painter and A. Spanias, "Perceptual coding of digital audio," in *Proc. IEEE*, Vol. 88, Issue 4, pp. 451-515, Apr. 2000.

[15] A. Aggarwal and K. Rose, "A conditional enhancement-layer quantizer for the scalable

MPEG advanced audio coder," in *Proc. ICASSP*, vol. 2, May 2002, pp. 1833-1836.

[16] I. Dimkoviae, et al., "Fast software implementation of MPEG advanced audio encoder," in *IEEE Int. Conf. Digital Signal Processing*, vol. 2, July 2002, pp. 839 – 843.

[17] http://www.audiocoding.com

[18] C. Y. Lee, et al., "Efficient AAC single layer transcoder," *AES 117th Convention*, San Francisco, Oct. 2004.

[19] Nakajima. Y, et al., "MPEG audio bit rate scaling on coded data domain," in *Proc. ICASSP*, vol. 6, May 1998, pp.3669-3672.

[20] Mat Hans, et al., "An MPEG audio layered transcoder," *AES 105th Convention*, San Francisco, Aug. 1998 (preprint 4812).

[21] European Broadcasting Union, "Sound Quality Assessment Material: Recordings for Subjective Tests, " Brussels, Belgium, Apr. 1988.

[22] Draft ITU-T Recommendation BS.1387 "Method for Objective Measurements of Perceived Audio Quality, " July 2001.

[23] A. Lerchs, "EAQUAL software, Version 0.1.3 alpha, " http://www.mp3-tech.org

簡　　歷

賴德亘：民國 1981 年生於台北市。 2004 畢業於台灣新竹的國立交通大學電子工程學系，之後進入該校電子工程所攻讀碩士學位。以音訊壓縮以及位元率轉碼器為論文研究主題。

Te-Hsueh Lai was born in Taipei in 1981. He received the BS degree in Department of Electronics Engineering, National Chiao Tung University (NCTU), HsinChu, Taiwan in 2004. His current research interests are audio transcoder and audio streaming.