Cascaded Trellis-Based Rate-Distortion Control Algorithm for MPEG-4 Advanced Audio Coding

Cheng-Han Yang and Hsueh-Ming Hang

***Abstract*—In this paper, a few low-complexity and high-performance rate-distortion control algorithms for MPEG-4 Advanced Audio Coding (AAC) are proposed. One key element in producing good quality compressed audio particularly at medium and low rates is a high performance rate-distortion controller in the audio encoder. Although the trellis-based rate-distortion control algorithms previously proposed can achieve a praiseworthy performance, their computational complexity is extremely high. Therefore, for practical applications, it is very desirable to achieve a similar performance at a much lower complexity. Two types of techniques are proposed in this paper to reduce the computational burden of the trellis-based algorithms. One is splitting a very heavy calculation stage into two sequential steps with much less computation. The other is reducing the candidates in the trellis for parameter search. Together, when applicable, our approach achieves a similar coding performance (audio quality) but requires less than 1/1000 complexity in computation.**

***Index Terms*—Advanced audio coding (AAC), audio coding, rate-distortion control, trellis-based search.**

I. Introduction

In THE last decade, analog audio has been gradually replaced by high-fidelity digital audio. Moreover, to meet the demand of efficient transmission and storage of digital audio for diversified multimedia applications, many high-efficient audio coding schemes have been developed, such as MPEG-1/2/4 audio coding standards and Dolby AC-3 [1]. The MPEG-4 advanced audio coding (AAC) is one of the most recent-generation audio coders specified by the ISO/IEC MPEG standards committee [2]. The core part of the MPEG-4 AAC is based on the MPEG-2 AAC technology. The MPEG-4 AAC features a number of additional coding tools and coder configurations comparing to MPEG-2 AAC [3], [4]. Consequently, the MPEG-4 AAC is a very efficient audio compression algorithm aiming at a wide variety of different applications, such as Internet, wireless, and digital broadcast arenas.

One critical element contributing to a good AAC encoder is selecting two sets of coding parameters properly, the scale factor (SF) and Huffman codebook (HCB) in the rate-distortion (R-D) loop. Because encoding these coding parameters is interband dependent in AAC, the proper choice of their values to maximize the coding performance becomes a difficult problem. Two-loop search (TLS) [5] is a commonly known R-D control algorithm, which is also used in the MPEG-4 AAC verification model (VM) [6]. VM is the encoder software developed by the MPEG committee to verify the coding syntax. However, as pointed out by [7] and [8], the poor choice of coding parameters in the TLS algorithm is one shortcoming of the current MPEG-4 AAC VM and, therefore, its compression efficiency is lower than expected particularly at low rates.

Two trellis-based high-performance R-D control algorithms for AAC are proposed by [7] and [8]. One distinct feature of these R-D control algorithms, as comparing to TLS, is that both bit rate and distortion are controlled simultaneously and the interband relationship of coding parameters, SF and HCB, is also counted in choosing their values. These R-D control algorithms are formulated as Viterbi search through the trellis diagram [9], [10] to find the optimal coding parameters and, therefore, are called *trellis-based optimization*. As discussed in [8], the subjective quality of the trellis-based optimization scheme is significantly better than that of TLS. However, its computational complexity is extremely high and thus it is not suitable for practical applications, such as real-time encoding with power constraint. Therefore, it is very desirable to achieve a similar performance at a much lower complexity.

In this paper, two types of techniques are introduced to speed up the trellis-based optimization procedure. In the first type of fast algorithms, we break the combined SF and HCB parameter selection stage into two sequential steps and thus call it *cascaded trellis-based* optimization. In the second type of fast algorithms, by observing the audio signal characteristics and statistics we develop a few rules that can reduce significantly the number of candidates in the trellis. These two techniques are fairly independent. Together, the overall computational complexity is dramatically reduced while the coding performance degradation is small.

The organization of this paper is as follows. In Section II, a brief overview of the typical MPEG-4 AAC encoder is provided. The proposed *cascaded trellis-based* R-D control algorithm and its variations are described in Section III. The proposed fast trellis search schemes are described in Section IV. The complexity analysis of the proposed R-D control algorithms and the simulation results with quality evaluation are summarized in Section V.

II. Overview on AAC Encoder

The block diagram of a typical MPEG-4 AAC encoder is shown in Fig. 1. The time-domain audio signals are first converted to their frequency-domain representation (spectral

Manuscript received April 11, 2004; revised February 25, 2005. This work was supported by the National Science Council, Taiwan, R.O.C., under Grant NSC-91-2219-E-009-011. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Ravi P. Ramachandran.

The Authors are with the Department of Electronics Engineering, National Chiao Tung University, Hsinchu 300, Taiwan, R.O.C. (e-mail: u8911831.ee89g@nctu.edu.tw; hmhang@mail.nctu.edu.tw).

Digital Object Identifier 10.1109/TSA.2005.857789

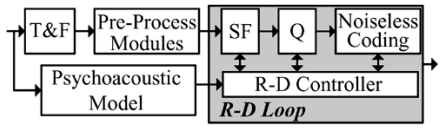

Fig. 1. Block diagram of a typical MPEG-4 AAC encoder. T&F: transform/filter bank, SF: scale factor, and Q: quantizer.

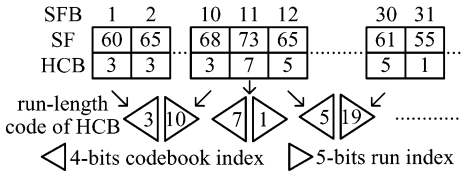

Fig. 2. Example of values of SF and HCB.

coefficients) by the modified discrete cosine transform. Motivated by the human auditory system, these spectral coefficients are grouped into a number of bands, called *scale factor bands* (SFB). The preprocessing modules, which are optional tools, can help enhancing the coding performance and enabling AAC to process a wide range of signals. The preprocessing modules in the MPEG-4 AAC include temporal noise shaping (TNS), long term prediction, intensity/coupling, prediction, perceptual noise substitution, and mid/side (M/S) stereo coding. The psychoacoustic model, its detailed procedure is outside the scope of the standards, calculates the hearing masking threshold, which is the base for deciding coding parameters in the R-D controller.

The spectral coefficients in one SFB are quantized by a nonuniform quantizer. The step size of the quantizer, which determines the quantization distortion (noise-to-masking ratio, NMR), is controlled by the parameter, SF. The quantized coefficients in one band are then entropy-coded by one of the twelve predesigned HCBs. Each SFB can have its own quantization step size and HCB. In addition, the indices of SFs and HCBs have to be coded and transmitted as side information. In AAC, the SFs are differentially coded relative to the previous SF and then Huffman coded using a predesigned codebook [2]. Taking Fig. 2 as example, instead of encoding the SF value of the second SFB, 65, the difference between the second SFB and the first SFB, 5, is coded. The indices of HCBs are coded by run-length codes [11]. A run-length code in AAC is nine bits long, which is composed of a four-bit codebook index and a five-bit run index. For example, as shown in Fig. 2, the third HCB is used from the first SFB to the 10th SFB; therefore, these ten HCB indices (same value) are coded together by one run-length code, in which the codebook index is 3 and run index is 10. The R-D controller, our focus in this paper, is to determine two critical parameters, the values of SF and HCB, for each SFB so as to optimize the selected criterion under the given bit rate constraint. In the following discussions, if the context is clear, the abbreviation "SF" is also referred to the value of SF and "HCB" is also referred to the index of HCB.

III. Cascaded Trellis-Based Optimization Scheme

In this section, we describe our first proposed fast algorithm, called *cascaded trellis-based* (CTB) scheme. We start with the problem formulation of the R-D control algorithm for AAC in Section III-A. The trellis-based (TB) procedures for SF optimization and HCB optimization in the CTB scheme are described in Sections III-B and III-C, respectively. One key element in the trellis-based optimization process on SF, so-called "pseudo-HCB" is explained in Section III-D. Finally, the procedure of the complete CTB optimization scheme is summarized in Section III-E.

A. Problem Formulation

For the perceptual audio coders, noise-to-masking ratio (NMR) is the most widely used objective measure in the R-D control module for modeling the subjective perceptual distortion. Based on NMR, there are two commonly used criteria for R-D optimization, the *average noise-to-mask ratio* (ANMR) and the *maximum noise-to-mask ratio* (MNMR) [12]. In AAC, the differential coding of SFs and the run-length coding of HCBs introduce interband dependence in parameter selection. In order to take into account the interband dependence in encoding SFs and HCBs, we need to consider all their possible combinations for all SFBs and examine the bits and distortion produced by each combination. If such interband dependence does not exist, we can decide SF and HCB for each SFB separately and add all bands together to find the global optimal solution.

Mathematically, the R-D optimization problems for minimizing ANMR and MNMR under a given bit rate constraint are formulated by (1) and (2), respectively,

$$\begin{aligned} &\text{minimize} \sum_i w_i d_i \text{ subject to (s.t.)} \\ &\sum_i (b_i + D(s_i - s_{i-1}) + R(h_{i-1}, h_i)) \le B \end{aligned} \tag{1}$$

$$\begin{aligned} &\text{minimize}(\max_i w_i d_i) \text{ subject to (s.t.)} \\ &\sum_i (b_i + D(s_i - s_{i-1}) + R(h_{i-1}, h_i)) \le B \end{aligned} \tag{2}$$

where i is the SFB index, w_i is the inverse of the masking threshold, and d_i is the quantization distortion, the mean squared quantization errors. In (1), $\sum_i w_i d_i$ is the sum of NMR over all SFBs in a frame and in (2), $\max_i w_i d_i$ is the maximum NMR in a frame. The parameter values of SF and HCB for the ith SFB are denoted by s_i and h_i, respectively. Symbol $D()$ is a function of SF, representing the number of bits produced by differential coding of SF. Symbol $R()$ is a function of HCB, representing the number of bits produced by run-length coding of HCB. The returned function values in both cases are numbers of bits to encode the arguments. Parameter b_i is the number of bits for coding the quantized spectral coefficients (QSCs) and the parameter B is the prescribed bit rate for an audio frame.

To solve (1) and (2), the straight forward joint optimization of SF and HCB for all SFBs is exorbitantly complex. For one frame in AAC, the number of SF values is 60, the number of HCB indices is 12, and there are 49 SFBs in total. Therefore, to find the optimal solution of all combinations, the complexity of brute force search is $O((60 \cdot 12)^{49})$. In [7] and [8], a dynamic programming approach, called *joint trellis-based* (JTB) scheme in this paper, is proposed to find the optimal SF and HCB for all

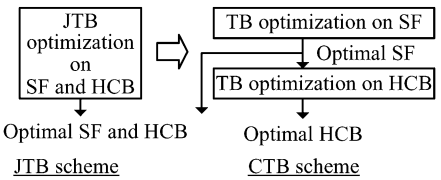

Fig. 3. Joint trellis-based scheme versus cascaded trellis-based scheme.

SFBs jointly at a reduced complexity. As shown in [7] and [8], the problem of minimizing ANMR in (1) can be reformulated as minimizing the unconstrained cost functions, C_{ANMR}, with the Lagrangian multiplier λ

$$C_{\mathrm{ANMR}} = \sum_i w_i d_i + \lambda \cdot (b_i + D(s_i - s_{i-1}) + R(h_{i-1}, h_i)). \tag{3}$$

Likewise, the problem of minimizing MNMR in (2) can be reformulated as minimizing the cost functions, C_{MNMR}, under the constraint: $w_i d_i \leq \lambda, \forall i$, for a certain value of λ

$$C_{\mathrm{MNMR}} = \sum_i b_i + D(f_i - f_{i-1}) + R(h_{i-1}, h_i). \tag{4}$$

The research in [7] and [8] shows that the problem of minimizing C_{ANMR} and C_{MNMR} can be efficiently solved by the Viterbi search through the trellis, in which we compute only the legal transitions from the previous state to the current state [9], [10]. Although, the search complexity of JTB scheme [8], $O((60 \cdot 12)^2 \cdot 49)$, is much lower than that of brute force search, it is still extremely high for practical applications.

As shown in Fig. 3, a simplification of the JTB scheme is to search for the SF and the HCB values in two consecutive steps without going through all possible combinations. Ideally, the order of complexity of our CTB scheme goes down to $O((60^2 + 12^2) \cdot 49)$. However, because these two steps are strongly correlated, we need to design the cascaded algorithm with special treatment on this issue to reduce performance degradation. This is the main point of this section.

B. Trellis-Based Optimization on SF

In this subsection, the procedures of trellis-based optimization on SF aiming at two criterions, ANMR and MNMR, are described separately in Sections III-B1 and III-B2.

1) Trellis-Based Procedure for ANMR Minimization: The problem of minimizing ANMR in the JTB scheme is formulated as minimizing the unconstrained cost functions, C_{ANMR}, in (3). However, to break the combined one step into two consecutive steps in our CTB scheme, this problem is reformulated as minimizing two unconstrained cost functions, $C_{\mathrm{SF\_ANMR}}$ and C_{HCB}, as follows:

$$C_{\mathrm{SF\_ANMR}} = \sum_i w_i d_i + \lambda \cdot (b_i + D(s_i - s_{i-1})) \tag{5}$$

$$C_{\mathrm{HCB}} = \sum_i b_i + R(h_{i-1}, h_i). \tag{6}$$

The minimization of $C_{\mathrm{SF\_ANMR}}$ is described in this subsection, and the minimization of C_{HCB} will be described in Section III-C. Because $C_{\mathrm{SF\_ANMR}}$ and C_{HCB} are minimized in two separate steps, the global optimality of C_{ANMR} is not guaranteed although the computation is significantly reduced. Our contribution described hereafter is to develop techniques that would come close to the global optimality.

Similar to the approach in the JTB scheme, the goal of finding proper SFs that minimize $C_{\mathrm{SF\_ANMR}}$ can be achieved by looking for the optimal path through the trellis. Each stage in the trellis corresponds to an SFB and there are N_SFB stages in total. However, different from JTB, each state at the ith stage in our scheme only represents an SF candidate for the ith SFB. In other words, at the ith stage, if a path passes through the mth state, it means that the mth SF candidate is used to encode the ith SFB.

For a given value of λ, the Viterbi search procedure for finding a proper set of SFs that minimize $C_{\mathrm{SF\_ANMR}}$ is outlined below. We denote $\mathrm{Y}_{k,i}$ as the kth state at the ith stage and denote $C_{k,i}$ as the minimum accumulative-partial cost ending at $\mathrm{Y}_{k,i}$. The state-transition cost, $T_{l,i-1 \to k,i}$, from $\mathrm{Y}_{l,i-1}$ to $\mathrm{Y}_{k,i}$ is $\lambda \cdot D(s_{k,i} - s_{l,i-1})$, where $s_{k,i}$ is the SF value associated with the state $\mathrm{Y}_{k,i}$.

1) Initialize all the states and start trellis search from the first stage. $C_{k,0} = 0, \forall k$ and $i = 1$.
2) For each state at the ith stage, find the best path from the previous stage by examining all the states at the $(i-1)$th stage leading to the current state. The best path ending at $\mathrm{Y}_{k,i}$ is the one that has the minimum accumulative-partial $C_{k,i}$. That is, we look for the minimum value of $C_{k,i}, \forall k$

$$C_{k,i} = \min_l \{C_{l,i-1} + (w_i d_{k,i} + \lambda \cdot b_{k,i} + T_{l,i-1 \to k,i})\}. \tag{7}$$

3) Check the index, i. If $i < N\_\mathrm{SFB}$, set $i = i + 1$ and go to step 2.

2) Trellis-Based Procedure for MNMR Minimization: The problem of minimizing MNMR in the JTB scheme is formulated as minimizing the cost functions, C_{MNMR}, in (4). In our CTB scheme, this problem is reformulated as the minimization of two cost functions, $C_{\mathrm{SF\_MNMR}}$ in (8) and C_{HCB} in (6), under the constraint: $w_i d_i \leq \lambda, \forall i$, for a certain value of λ

$$C_{\mathrm{SF\_MNMR}} = \sum_i b_i + D(s_i - s_{i-1}). \tag{8}$$

Similar to the trellis-based ANMR optimization on selecting SF described above, an "SF trellis" is constructed for minimizing $C_{\mathrm{SF\_MNMR}}$. For a given value of λ, the Viterbi search procedure for finding proper SFs that minimize $C_{\mathrm{SF\_MNMR}}$ is outlined below. The state-transition cost, $T_{l,i-1 \to k,i}$, is $D(s_{k,i} - s_{l,i-1})$.

1) Initialize. $C_{k,0} = 0, \forall k$ and $i = 1$.
2) For the ith stage, only the particular state, which the $\mathrm{NMR}(w_i d_{k,i})$ associated with is less than or equal to λ, is valid for trellis search. Therefore, before staring the trellis search, we must find the valid states for the ith stage, $\mathrm{Y}_{k,i}, \forall k$.
3) For each valid state at the ith stage, find the best path from the previous stage by examining all the valid states in the $(i-1)$th stage leading to the current state. That is, we compute and find the $C_{k,i}$ such that

$$C_{k,i} = \min_l \{C_{l,i-1} + (b_{k,i} + T_{l,i-1 \to k,i})\} \tag{9}$$

4) If $i < N\_\mathrm{SFB}$, set $i = i + 1$ and go to step 2.

After completing the forward "search and expansion" step through the trellis, the optimal path in the trellis can be extracted by tracing backward from the state with minimum C_{k,N_SFB} at the last stage. Consequently, the optimal SFs for all SFBs that minimize $C_{\mathrm{SF\_MNMR}}$ (or $C_{\mathrm{SF\_ANMR}}$) are determined.

As described in [7] and [8], to a band below the masking threshold, any values of SF can be assigned. Therefore, its associated state in the trellis is split into two consecutive states. At the first state, the spectral coefficients are quantized using the assigned valid SF, and at the second state, all quantized values of spectral coefficients are set to zero.

C. Trellis-Based Optimization on HCB

The HCB optimization is performed under the assumption that the SF (value) for each SFB has already been decided. In our CTB scheme, SF is determined by the trellis-based optimization on SF described in Section III-B. With a determined SF, QSCs for each SFB are fixed and thus the b_i term in the cost function C_{HCB} [see (6)] depends only on the selection of HCB. Therefore, C_{HCB} can be restated as

$$C_{\mathrm{HCB}} = \sum_i H_{h_i}(\boldsymbol{q}_i) + R(h_{i-1}, h_i) \tag{10}$$

where $\boldsymbol{q}_i$ (vector) contains the QSCs for the ith SFB and symbol $H_h()$ is a function of QSCs, representing the number of bits produced by Huffman-coding of QSCs using the hth HCB. The goal of the optimization procedure here is to find the HCBs for all SFBs that minimize the cost function C_{HCB} and this can be achieved again by finding the optimal path through the trellis with states now being HCB.

An "HCB trellis" is thus constructed for searching for the minimum C_{HCB}. Each state at the ith stage represents an HCB candidate for the ith SFB. The state-transition cost, $T_{n,i-1\to m,i}$, from $\mathrm{Y}_{n,i-1}$ to $\mathrm{Y}_{m,i}$ is $R(h_{n,i-1}, h_{m,i})$, where $h_{m,i}$ is the HCB associated with the state $\mathrm{Y}_{m,i}$. According to the run-length coding rule in AAC, $R(h_{n,i-1}, h_{m,i})$ is defined by (11). In other words, no extra bits are transmitted if the same HCBs are used in two neighboring SFBs

$$R(h_{n,i-1}, h_{m,i}) = \begin{cases} 0, & \text{if } n = m \\ 9, & \text{otherwise} \end{cases}. \tag{11}$$

The Viterbi search procedure for finding proper HCBs that minimize C_{HCB} is outlined below.

1) Initialize. $C_{m,0} = 0, \forall m$ and $i = 1$.
2) For each state at the ith stage, find the best path from the previous stage by examining all the states at the $(i-1)$th stage leading to the current state. That is, we find the best $\mathrm{Y}_{m,i}$ by computing and find the $C_{m,i}$ such that

$$C_{m,i} = \min_n \{C_{n,i-1} + (H_{h_{m,i}}(\boldsymbol{q}_i) + T_{n,i-1\to m,i})\} \tag{12}$$

where $\boldsymbol{q}_i$ is the vector of the QSCs in the stage i.
3) If $i < N\_\mathrm{SFB}$, set $i = i + 1$ and go to step 2.

Similar to the trellis-based optimization on SF, after completing the forward search/expansion step through the trellis, the optimal path in the trellis can be extracted by tracing backward from the minimum C_{m,N_SFB} state at the last stage. Then, the optimal HCBs for all SFBs that minimize C_{HCB} are determined.

D. Pseudo HCB for SF Optimization

1) Motivation for Pseudo HCB: We first look at the MNMR minimization case. The key problem in splitting (4) into (8) and (6) is to choose the correct (optimal) value of $b_{k,i}$ in (8). In (8), the $w_i d_{k,i}$ or $D(s_{k,i} - s_{l,i-1})$ term is unique for a given state or state transition in the SF trellis. However, the value of $b_{k,i}$ depends not only on $s_{k,i}$ associated with the state in the SF trellis; it also depends on the choice of HCB. In the JTB scheme, SF and HCB are chosen simultaneously. Therefore, for each candidate value of SF, all possible $b_{k,i}$ values, corresponding to 12 candidate HCBs, are evaluated. In other words, the chosen value of $b_{k,i}$ for each state $\mathrm{Y}_{k,i}$ in the trellis for JTB optimization scheme is optimal [7], [8]. But in our sequential optimization scheme, the value of $b_{k,i}$ for the state $\mathrm{Y}_{k,i}$ in (8) is estimated based upon the available information. The estimated value of $b_{k,i}$ may not be the optimal value and this may further induce an incorrect (nonoptimal) selection in SF optimization. For example, two candidate paths in the SF trellis, A and B, are shown in (13). Path A is better than path B because $C^{\mathrm{A}}_{\mathrm{SF\_MNMR}} < C^{\mathrm{B}}_{\mathrm{SF\_MNMR}}$, where $C^{\mathrm{A}}_{\mathrm{SF\_MNMR}}$ and $C^{\mathrm{B}}_{\mathrm{SF\_MNMR}}$ are the $C_{\mathrm{SF\_MNMR}}$ values of path A and path B, respectively. Note that $\hat{b}^{\mathrm{A}}_i$ and $\hat{b}^{\mathrm{B}}_i$ in (13) are the estimated values of b_i for path A and path B. If the decision on SF is made at this point, path A is chosen. Now, let us go one step further. Based on the selected SF sets of path A and path B, we can find their optimal HCBs, h^{A}_i and h^{B}_i respectively, according to the HCB optimization procedure described in Section III-C. Then, their actual bits information b^{A}_i and b^{B}_i, for path A and path B, respectively, is obtained. Finally, the total costs $C^{\mathrm{A}}_{\mathrm{MNMR}}$ and $C^{\mathrm{B}}_{\mathrm{MNMR}}$ for two candidate paths are shown in (14). The result in (14) indicates that path B is actually better than path A when the bits information is correct. With a wrong estimate on b_i, our CTB algorithm would pick up path A for SFs and thus it fails to find the overall optimal path B

$$\begin{aligned} C^{\mathrm{A}}_{\mathrm{SF\_MNMR}} &= \sum_i \hat{b}^{\mathrm{A}}_i + D\left(s^{\mathrm{A}}_i - s^{\mathrm{A}}_{i-1}\right) \\ &< C^{\mathrm{B}}_{\mathrm{SF\_MNMR}} = \sum_i \hat{b}^{\mathrm{B}}_i + D\left(s^{\mathrm{B}}_i - s^{\mathrm{B}}_{i-1}\right) \end{aligned} \tag{13}$$

$$\begin{aligned} C^{\mathrm{A}}_{\mathrm{MNMR}} &= \sum_i \left(b^{\mathrm{A}}_i + D\left(s^{\mathrm{A}}_i - s^{\mathrm{A}}_{i-1}\right) + R\left(h^{\mathrm{A}}_{i-1}, h^{\mathrm{A}}_i\right)\right) \\ &> C^{\mathrm{B}}_{\mathrm{MNMR}} = \sum_i \left(b^{\mathrm{B}}_i + D\left(s^{\mathrm{B}}_i - s^{\mathrm{B}}_{i-1}\right)\right. \\ &\quad \left. + R\left(h^{\mathrm{B}}_{i-1}, h^{\mathrm{B}}_i\right)\right). \end{aligned} \tag{14}$$

Clearly, with a more accurate estimate on $b_{k,i}$, we can select better SFs. For this aim, the concept of "pseudo HCB" is proposed for the trellis-based optimization on SF. The preceding discussions on choosing HCB can be applied to the ANMR minimization case.

2) Design of Pseudo HCB: When the trellis-based optimization on SF is performed in the pseudo HCB mode, a pseudo HCB with an index set $h^v_{k,i}$ needs to be constructed for the state $\mathrm{Y}_{k,i}$ to produce $b_{k,i}$ in (5) and (8). It can be constructed in several ways. For example, $h^v_{k,i}$ may contain only one of the 12 candidate HCBs or several codebooks. In order to improve the accuracy of the estimated values of $b_{k,i}$ and $h^v_{k,i}$, we analyze the data collected from the JTB optimization scheme.

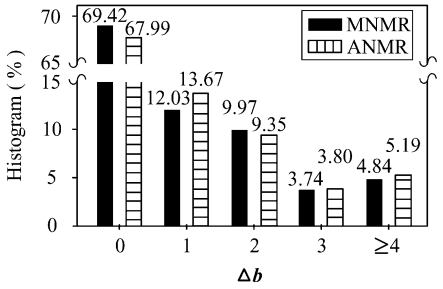

Fig. 4. Histogram on $\Delta \boldsymbol{b}$.

For a given value of λ, using the JTB scheme, we can find a set of optimal parameters, $\boldsymbol{s}_{\text{opt}}^{\text{JTB}}$, $\boldsymbol{h}_{\text{opt}}^{\text{JTB}}$, and $\boldsymbol{b}_{\text{opt}}^{\text{JTB}}$ that minimizes the cost function, C_{ANMR} in (3) or C_{MNMR} in (4). For comparison purposes, we also construct a reference set of QSC bits, $\boldsymbol{b}_{\text{min}}^{\text{JTB}}$, in the following way. For the ith SFB, $b_{\text{min},i}^{\text{JTB}}$ is the minimum number of bits for encoding $\boldsymbol{q}_{\text{opt},i}^{\text{JTB}}$ and is determined by $b_{\text{min},i}^{\text{JTB}} = \min_m\{H_m(\boldsymbol{q}_{\text{opt},i}^{\text{JTB}})\}$, where $q_{\text{opt},i}^{\text{JTB}}$ is the QSCs quantized by using $s_{\text{opt},i}^{\text{JTB}}$. In other words, without considering the bits for coding the HCB indices, $b_{\text{min},i}^{\text{JTB}}$ is the lowest bits number produced by any of the 12 HCBs applied to the QSCs. Because the coded bits for HCB indices, $R(h_{i-1}, h_i)$, are also included in the overall optimization procedure, when comparing coding bits for QSCs only, $\boldsymbol{b}_{\text{opt}}^{\text{JTB}}$ is higher than or equal to $\boldsymbol{b}_{\text{min}}^{\text{JTB}}$.

By collecting the statistics from the simulations on ten audio sequences, the histogram of the differences between $\boldsymbol{b}_{\text{opt}}^{\text{JTB}}$ and $\boldsymbol{b}_{\text{min}}^{\text{JTB}}$, denoted by $\Delta \boldsymbol{b}$, is shown in Fig. 4. We observe that over 91% of $\Delta \boldsymbol{b}$ is less than 3 for both ANMR and MNMR criterions. In general, we can choose the HCB that produces the minimum QSC bits, $\boldsymbol{b}_{\text{min}}^{\text{JTB}}$.

After examining this characteristics of $\boldsymbol{b}_{\text{opt}}^{\text{JTB}}$, we derive a rule in determining $h_{k,i}^v$ and $b_{k,i}$. For the state $\text{Y}_{k,i}$, $h_{k,i}^v$ is the index set of HCB that satisfies the proposed rule in (15), namely

$$h_{k,i}^v = \{n \mid H_n(\boldsymbol{q}_{k,i}) \le \min_m\{H_m(\boldsymbol{q}_{k,i})\} + \delta, \quad n \in \{0, \ldots, 11\}\}. \tag{15}$$

The $\min_m\{H_m(\boldsymbol{q}_{k,i})\}$ term is the minimum number of bits for coding $\boldsymbol{q}_{k,i}$ without considering the coding bits for HCB indices and δ is an offset parameter. For example, if $H_1(\boldsymbol{q}_{k,i})$ and $H_3(\boldsymbol{q}_{k,i})$ are both smaller than or equal to $\min_m\{H_m(\boldsymbol{q}_{k,i})\}+\delta$, then $h_{k,i}^v$ equals to $\{1, 3\}$. Although (5) and (8) do not include the bits number for coding HCB indices, it is found from experiments that including this term leads to a better estimate of SF. Therefore, we expand (5) to approximate (3) and expand (8) to approximate (4) with additional terms. Based on the above observation, $b_{k,i}$ is rewritten as

$$b_{k,i} = \left(\left(\sum_{n \in h_{k,i}^v} H_n(q_{k,i}) \right) \Big/ \left| h_{k,i}^v \right| \right) + \alpha \cdot R_v\left(h_{l,i-1}^v, h_{k,i}^v\right) \tag{16}$$

where α is a weight for including $R_v(h_{l,i-1}^v, h_{k,i}^v)$ into $b_{k,i}$ and $|h_{k,i}^v|$ is the number of elements in $h_{k,i}^v$.

The symbol R_v is the run-length coding function performed on the pseudo HCB and is defined below

$$R_v\left(h_{l,i-1}^v, h_{k,i}^v\right) = \begin{cases} 0, & \text{if } (h_{l,i-1}^v \cap h_{k,i}^v) \neq \phi \\ 9, & \text{otherwise.} \end{cases} \tag{17}$$

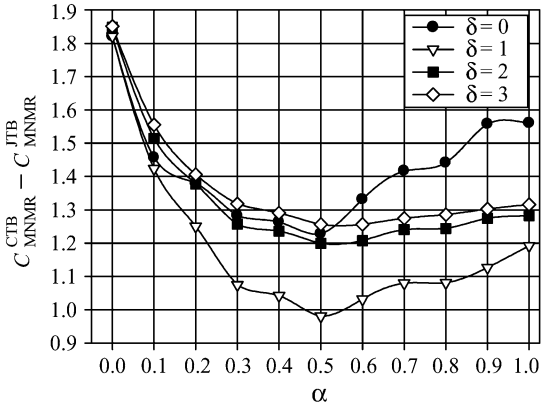

Fig. 5. $(C_{\text{MNMR}}^{\text{CTB}} - C_{\text{MNMR}}^{\text{JTB}})$ versus (δ, α).

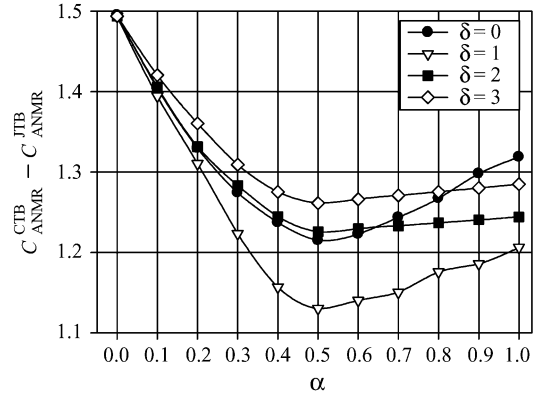

Fig. 6. $(C_{\text{ANMR}}^{\text{CTB}} - C_{\text{ANMR}}^{\text{JTB}})$ versus (δ, α).

Note that the $R_v()$ function is essentially the $R()$ function in (11). However, because $h_{k,i}^v$ and $h_{l,i-1}^v$ are index sets of HCB, the intersection is used in (17).

After having derived (15) and (16), we still need to determine the proper values for δ and α. The values of δ and α can be determined by examining the difference between the JTB scheme and the CTB scheme at different values of δ and α and the results are shown in Figs. 5 and 6. Note that $C_{\text{ANMR}}^{\text{JTB}}$ and $C_{\text{ANMR}}^{\text{CTB}}$ are the C_{ANMR} [in (3)] derived using the JTB scheme and CTB scheme, respectively. $C_{\text{MNMR}}^{\text{JTB}}$ and $C_{\text{MNMR}}^{\text{CTB}}$ are the C_{MNMR} [in (4)] derived using the JTB scheme and CTB scheme, respectively. We find that for a wide range of δ values, we can achieve a pretty good performance when $R_v(h_{l,i-1}^v, h_{k,i}^v)$ is included in $b_{k,i}(\alpha > 0)$. As Figs. 5 and 6 indicate, the case that $\delta = 1$ and $\alpha = 0.5$ gives the best results. Hence, we choose 1 for δ and 0.5 for α in our implementation.

E. Cascaded Trellis-Based Optimization Procedure

The major steps in our CTB scheme have been described in detail in Sections III-B to III-D. The flowchart of the complete CTB optimization scheme is summarized in Fig. 7. Passing through the first step (block), we obtain a set of optimal SF, $\boldsymbol{s}_{\text{opt}}$. Then, the second step produces $\boldsymbol{h}_{\text{opt}}$, a set of optimal HCB. Based on this set of $\boldsymbol{h}_{\text{opt}}$, $\boldsymbol{s}'_{\text{opt}}$ is a new set of optimal SF derived at the end of the third step. Note that the same trellis-based optimization on SF is used in steps 1 and 3, but they are derived using different HCB modes. In step 1,

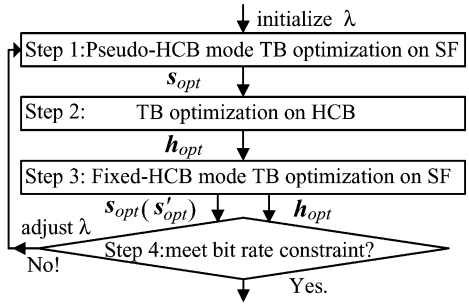

Fig. 7. Flowchart of the cascaded trellis-based optimization scheme.

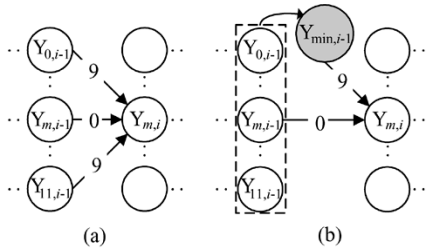

Fig. 8. Trellis representation of the HCB transition.

the pseudo-HCB mode is used and in step 3 the fixed-HCB mode is used. The pseudo-HCB mode has been described in Section III-D. For the fixed-HCB mode, an index set of fixed HCBs, $\boldsymbol{h}^X = [h_1^X \ h_2^X \ \ldots \ h_{N\_\mathrm{SFB}}^X]$, is prechosen and used in the trellis-based SF optimization procedure. For all the states at the ith stage in the SF trellis in Section III-B, the QSCs, $\boldsymbol{q}_{k,i}$, are entropy-coded using the h_i^Xth HCB; therefore, the value of $b_{k,i}$ in (5) and (8) is correctly calculated by $b_{k,i} = H_{h_i^X}(\boldsymbol{q}_{k,i}), \forall k$. In this flowchart, $\boldsymbol{h}^X$ is derived from step 2 and is the final $\boldsymbol{h}_{\mathrm{opt}}$ in our CTB scheme.

The four-step procedure in Fig. 7 is called *two-iteration mode*, because the optimization process on SF is done twice. The second optimization step on SF (step 3) can recover some inadequate SF values determined in step 1 owing to the incorrect HCB used in the pseudo HCB model. The CTB optimization can be further simplified, in which step 3 is omitted to save computation. This is called *one-iteration mode*. Clearly, there is a trade-off between complexity and coding performance.

IV. Fast Trellis Search Algorithm

As described in the previous section, the basic structure of our CTB algorithm (or JTB algorithm) is trellis search. If we can reduce the complexity of trellis search, we speed up the entire process. In this section, we propose fast algorithms aiming at reducing the trellis search complexity. The complexity of the trellis-based optimization scheme depends on the searching range (number of states) of each stage in the trellis. Hence, reducing the candidate states at each stage is an effective way in reducing the overall computational complexity.

A. Fast Search Algorithm for HCB Optimization

In AAC, SFs are differentially coded and HCBs are coded by run-length coding. Run-length coding can be viewed as a special case of differential coding; therefore, the procedure of trellis-based optimization on HCB is similar to that on SF. However, the output of run-length coding has only two possible values, either 0 or 9 as shown in (11). As shown in Fig. 8(a), in order to find the optimal path ending at $\mathrm{Y}_{m,i}$, all the HCB candidates at the $(i-1)$th stage have to be taken into consideration. In AAC, there are 12 predesigned HCBs; thus, the searching complexity for finding all the optimal paths ending at the ith stage is 12×12.

The number on the directional branch in Fig. 8 is the state-transition cost. Except for the path $\mathrm{Y}_{m,i-1} \rightarrow \mathrm{Y}_{m,i}$, the state-transition costs of the other 11 paths ending at $\mathrm{Y}_{m,i}$ are all identical (equal to 9). Therefore, in calculating $C_{m,i}$ in (12), among these 11 paths, the path with the least $C_{n,i-1}$ will result in the smallest $C_{m,i}$. Based on this property, a fast search algorithm is proposed, which is divided into two steps.

1) Among the 12 candidate states at the $(i-1)$th stage, the state with the minimum cost, $C_{\min,i-1}$, is chosen and treated as the pseudo thirteenth state, $\mathrm{Y}_{\min,i-1}$, and $C_{\min,i-1} = \min_n\{C_{n,i-1}\}$.
2) As shown in Fig. 8(b), when finding the optimal path ending at $\mathrm{Y}_{m,i}$, we only have to consider two paths, path $(\mathrm{Y}_{m,i-1} \rightarrow \mathrm{Y}_{m,i})$ and path $(\mathrm{Y}_{\min,i-1} \rightarrow \mathrm{Y}_{m,i})$. The rest of this revised searching procedure is the same as that in Section III-C.

The searching complexity (in terms of branch metric calculation) of this fast algorithm is approximately $12 + 2 \times 12$, which is about $1/4$ of the complexity of the full search algorithm. The first "12" term is needed for determining $\mathrm{Y}_{\min,i-1}$. Note that the performance (accuracy) of this fast search algorithm is the same as that of the full search algorithm. Therefore, this fast search algorithm can be used by both CTB and JTB optimization schemes without any performance loss.

B. Fast Search Algorithm for SF Optimization (MNMR)

In the trellis-based optimization on SF, each state in the trellis represents an SF candidate. Searching over a larger set of SF candidates can result in better performance, but at the cost of higher searching complexity. In general, the number of states (sn) for all the stages in the trellis are the same and the searching complexity for each stage transition in this uniform trellis is $\mathrm{sn} \times \mathrm{sn}$.

In this section, we propose two nonuniform (adaptive) trellis search algorithms for SF optimization under MNMR criterion, in which the number of state for each stage in the trellis can vary to reduce the overall searching complexity. The first one is called "**G**lobal **m**inimum (reference) SF-restricted **N**on-**u**niform *trellis*," or "GMNU" in short, and the second one is called "**L**ocal **m**inimum (reference) SF-restricted **N**on-**u**niform *trellis*," or "LMNU." In both cases, a reference SF is first identified and then the number of candidates is reduced against this reference. Note that all SFs at the ith stage in the trellis are sorted and indexed in ascending order.

First of all, we define the reference SF, s_i^{ref}, for the ith SFB as the largest SF among all the valid states at the ith stage. Note that s_i^{ref} is the upper bound of SF candidate at the ith stage, which means that the SF values of all the other valid states at the ith stage are less than s_i^{ref}. In the GMNU algorithm, we

define the integer index of the "global minimum" (reference) SF, $s^{\mathrm{ref}}_{G\cdot\mathrm{Min}}$, as the minimum reference SF value of *all* the scale factor bands; that is, $s^{\mathrm{ref}}_{G\cdot\mathrm{Min}} = \min_j\{s^{\mathrm{ref}}_j\}$. Then, we restrict the SF candidates at the ith stage in the range of $[s^{\mathrm{ref}}_{G\cdot\mathrm{Min}} - \varepsilon, s^{\mathrm{ref}}_i]$. Namely, we only use the SF values between s^{ref}_i and $s^{\mathrm{ref}}_{G\cdot\mathrm{Min}} - \varepsilon$. Thus, the number of state at the ith stage, $\mathrm{sn}_{\mathrm{Gm},i}$, equals to $(s^{\mathrm{ref}}_i - s^{\mathrm{ref}}_{G\cdot\mathrm{Min}} + 1 + \varepsilon)$.

In the LMNU algorithm, we define the integer index of the ρth-order "local minimum" (reference) SF at the ith stage, $s^{\mathrm{ref}}_{L\cdot\mathrm{Min},i}$, as $s^{\mathrm{ref}}_{L\cdot\mathrm{Min},i} = \min_{i-\rho\le j\le i+\rho}\{s^{\mathrm{ref}}_j\}$. Essentially, instead of all the bands, we only look at a *local neighborhood* of the current stage. Then, we restrict the SF candidates at the ith stage in the range of $[s^{\mathrm{ref}}_{L\cdot\mathrm{Min},i} - \varepsilon, s^{\mathrm{ref}}_i]$. Therefore, the number of states for the ith stage, $\mathrm{sn}_{\mathrm{Lm},i}$, equals to $(s^{\mathrm{ref}}_i - s^{\mathrm{ref}}_{L\cdot\mathrm{Min},i} + 1 + \varepsilon)$. In both cases, ε is the parameter that controls the search range. In the simulations in Section V, the value of ε is set to 1 and the 1th-order $(\rho = 1)$ local minimum reference SF is used in the LMNU algorithm.

We first explain our motivation behind the fast GMNU algorithm. As shown in (8) in Section III-B, The cost function $C_{\mathrm{SF\_MNMR}}$ can be divided into two parts, the differentially coded bits of SF values, $\Sigma_i D(s_i - s_{i-1})$, and the QSC coded bits, $\Sigma_i b$. In general, a larger value of SF will result in a smaller value of QSC and thus fewer coded QSC bits. If we set the SF value of the ith SFB to $s^{\mathrm{ref}}_i, \forall i$, we achieve the globally minimal $\Sigma_i b$ but this rule leads to a larger $\Sigma_i D(s_i - s_{i-1})$. On the other hand, if we set the SF values of all SFBs to $s^{\mathrm{ref}}_{G\cdot\mathrm{Min}}$, we achieve the globally minimal $\Sigma_i D(s_i - s_{i-1})$ because the differential SF values are all zero, but this rule leads to a larger $\Sigma_i b$. Therefore, a good guess is that the optimal SF value that minimizes $C_{\mathrm{SF\_MNMR}}$ likely falls in the range of $[s^{\mathrm{ref}}_{G\cdot\mathrm{Min}}, s^{\mathrm{ref}}_i]$. Although exceptions could exist, our guess by far dominates. The statistics show that the probability of occurrence of exceptions is less than 0.5% and the average increased value on $C_{\mathrm{SF\_MNMR}}$ due to exceptions is less than 1 bit.

The idea behind the LMNU algorithm is similar. However, we only look at the local $\Sigma_i b$ and $\Sigma_i D(s_i - s_{i-1})$ values in this case. Therefore, the LMNU algorithm requires an even lower computation but it leads to a higher distortion. Note that, GMNU and LMNU algorithms also can be used by both CTB and JTB optimization schemes.

As for the trellis-based ANMR optimization on SF, the cost function $C_{\mathrm{SF\_ANMR}}$ depends not only on $\Sigma_i b$ and $\Sigma_i D(s_i - s_{i-1})$ but also on $\Sigma_i w_i d_i$. Therefore, the GMNU or LMNU fast search algorithm cannot be applied.

V. Simulation Results

In this section, we will evaluate the computational complexity and the coded audio quality of our proposed algorithms. Four types of R-D control algorithms are simulated and compared as described below.

1) The TLS algorithm in MPEG-4 AAC VM (VM-TLS).
2) The JTB optimization schemes aiming at minimizing ANMR and MNMR, abbreviated as JTB-ANMR and JTB-MNMR respectively, proposed in [7] and [8].
3) The CTB optimization schemes aiming at minimizing ANMR and MNMR, abbreviated as CTB-ANMR and CTB-MNMR respectively, described in Section III.
4) The CTB-MNMR (or JTB-MNMR) incorporating GMNU, LMNU and the fast search algorithm for HCB optimization (FSHCB) described in Section IV.

TABLE I
Complexity Analysis for JTB, CTB, and Fast Search Algorithms

Scheme	Computation	Ratio	Storage
JTB-ANMR (JTB-MNMR)	$(60 \times 2)^2 \times 12^2$	1	120 x 12
CTB-ANMR (CTB-MNMR)	$n \times (60 \times 2)^2 + 12^2$	n /142	120
JTB-MNMR+ GMNU+FSHCB	$(sn^{ave}_{Gm} \times 2)^2 \times 36$	~1/(25 x 4)	120 x 12
JTB-MNMR+ LMNU+FSHCB	$(sn^{ave}_{Lm} \times 2)^2 \times 36$	~1/(144 x 4)	120 x 12
CTB-MNMR+ GMNU+FSHCB	$n \times (sn^{ave}_{Gm} \times 2)^2 + 36$	n /3600	120
CTB-MNMR+ LMNU+FSHCB	$n \times (sn^{ave}_{Lm} \times 2)^2 + 36$	$(n+0.4)/20736$	120

To focus only on the R-D performance, all the optional tools in AAC, such as TNS and M/S stereo coding, are not used in our simulations. Ten two-channel audio sequences with a sampling rate at 44.1 kHz are tested. Two of them are extracted from MPEG SQAM [6], and the rest are from EBU [13].

A. Complexity Analysis

The complexity analysis for the aforementioned several R-D control algorithms is summarized in Table I. The "Computation" column is the number of branch metrics in calculating one-stage transition in the trellis. For the convenience of comparison, the JTB-ANMR or JTB-MNMR is chosen to be the reference $(\mathrm{ratio} = 1)$ and all the other schemes are rated based on this reference. Also shown in Table I is the storage requirement. Again, it is measured in the number of branch metrics.

We can find from Table I that the CTB-ANMR and CTB-MNMR schemes are approximately $(142/n)$ times faster than the JTB-ANMR and JTB-MNMR schemes, in which n equals to 1 or 2. Moreover, the storage requirement for the trellis search in the CTB-ANMR and CTB-MNMR schemes is much smaller than that in the JTB-ANMR and JTB-MNMR scheme.

For the JTB scheme, the fast HCB search algorithm can be adopted to reduce the complexity down to 1/4. Note that $\mathrm{sn}^{\mathrm{ave}}_{\mathrm{Gm}}$ and $\mathrm{sn}^{\mathrm{ave}}_{\mathrm{Lm}}$ in Table I are the average number of states in the GMNU and LMNU algorithms and are calculated by using the following:

$$\mathrm{sn}^{\mathrm{ave}}_{\mathrm{Gm}} = \left(\sum_{i=1}^{N\_\mathrm{SFB}} (\mathrm{sn}_{\mathrm{Gm},i-1}\cdot \mathrm{sn}_{\mathrm{Gm},i}) \Big/ N\_\mathrm{SFB}\right)^{1/2} \quad (18)$$

$$\mathrm{sn}^{\mathrm{ave}}_{\mathrm{Lm}} = \left(\sum_{i=1}^{N\_\mathrm{SFB}} (\mathrm{sn}_{\mathrm{Lm},i-1}\cdot \mathrm{sn}_{\mathrm{Lm},i}) \Big/ N\_\mathrm{SFB}\right)^{1/2}. \quad (19)$$

The simulation data show that a typical $\mathrm{sn}^{\mathrm{ave}}_{\mathrm{Gm}}$ is approximately 12 and $\mathrm{sn}^{\mathrm{ave}}_{\mathrm{Lm}}$ is around 5. Hence, the GMNU algorithm can re-

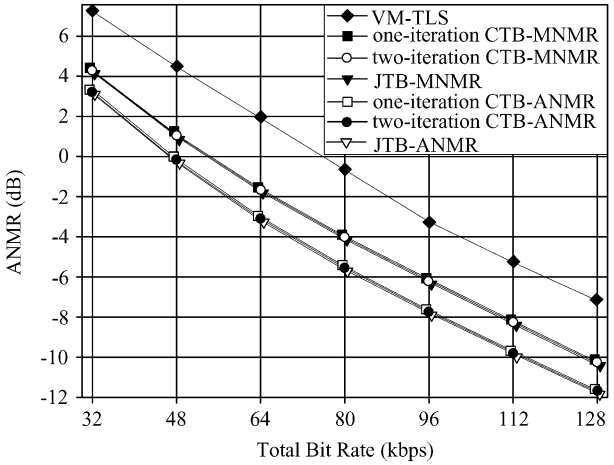

Fig. 9. ANMR rate-distortion comparison for VM-TLS, JTB, and CTB.

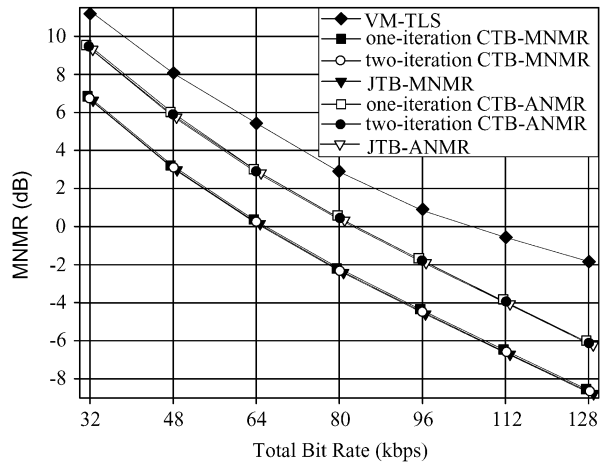

Fig. 10. MNMR rate-distortion comparison for VM-TLS, JTB, and CTB.

duce the complexity to $(12/60)^2 = 1/25$ and the LMNU algorithm can reduce the complexity to $(5/60)^2 = 1/144$.

B. Objective Quality

The rate-distortion curves of these bit allocation schemes are displayed in Figs. 9 and 10. Two major distortion metrics, ANMR and MNMR, are in use. We can find that the performance of the CTB scheme is similar to that of the JTB scheme. The ANMR performance loss is less than 0.2 dB for the one-iteration CTB-ANMR and less than 0.1 dB for the two-iteration CTB-ANMR (the lowest three curves in Fig. 9). The MNMR performance loss is less than 0.1 dB for both one- and two-iteration CTB-MNMR (the lowest three curves in Fig. 10). All of them are much better than the VM-TLS (the top line).

The differences of performance between the fast SF search algorithms and the full search (original) algorithm for the CTB-MNMR scheme are shown in Figs. 11 and 12. Note that the original CTB-MNMR scheme uses the uniform trellis with the state number $\mathrm{sn} = 60$ in SF optimization. In addition to the two nonuniform trellis fast algorithms, GMNU and LMNU, for comparison purpose, we create two *uniform* trellis with smaller

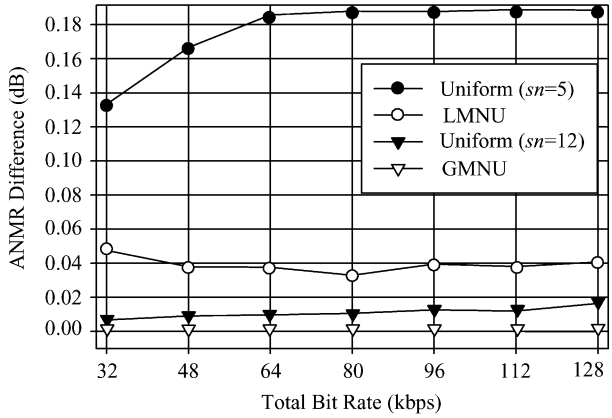

Fig. 11. ANMR differences between the full search and the fast SF search algorithms for CTB-MNMR.

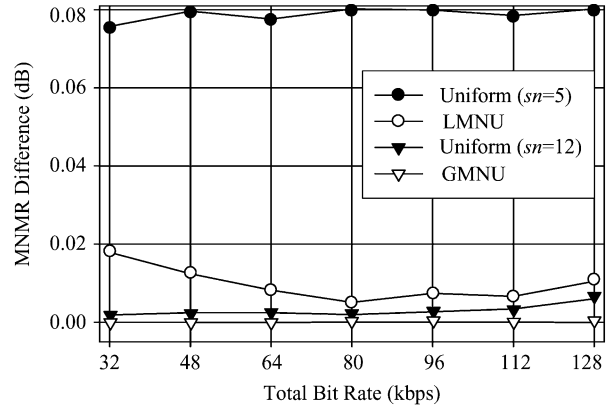

Fig. 12. MNMR differences between the full search and the fast SF search algorithms for CTB-MNMR.

numbers of states, namely, $\mathrm{sn} = 5$ and $\mathrm{sn} = 12$, which approximately equal to the values of $\mathrm{sn}_{\mathrm{Gm}}^{\mathrm{ave}}$ and $\mathrm{sn}_{\mathrm{Lm}}^{\mathrm{ave}}$ in Section V-A. There is nearly no performance loss for the GMNU algorithm $(\mathrm{ANMR\ or\ MNMR\ Difference} \approx 0)$. The penalty on LMNU is small but exists. The advantage of the nonuniform algorithms over the uniform algorithms at about the same complexity is clearly shown in Figs. 11 and 12.

C. Subjective Quality

Listening test by human ears is the traditional way to subjectively evaluate the audio quality and is also the most recognized subjective quality measure. However, such subjective test is expensive, time consuming, and difficult to reproduce. As described in in [8, Sec VI-B], the subjective quality (mean opinion score, MOS) of the JTB-ANMR (or JTB-MNMR) scheme is significantly better than that of the VM-TLS. MOS is derived from the ITU (International Telecommunications Union) five-grade absolute category rating (ACR) scheme [14]. Moreover, the informal listening tests on the aforementioned schemes show that it is hard to tell the difference between JTB and various CTB schemes. In addition, a "*simulated*" subjective measure, Objective Difference Grade (ODG), is used in audio quality evaluation. ODG is generated by a procedure designed to be comparable to the Subjective Difference Grade (SDG) judged by human ears. It is calculated based on the difference between the

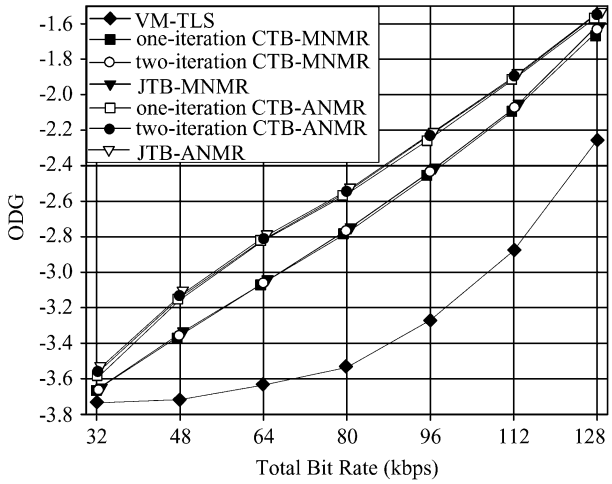

Fig. 13. ODG of VM-TLS, JTB, and CTB.

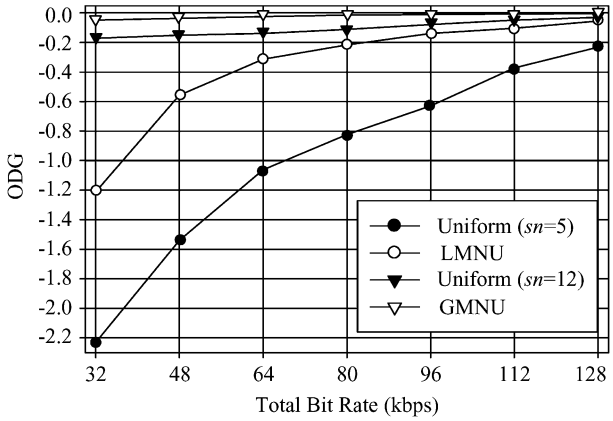

Fig. 14. ODG of various fast SF search algorithms for CTB-MNMR.

quality rating of the "reference" signal and the "test" signal. The ODG has a range of $[-4, 0]$, in which -4 stands for very significant difference and 0 stands for imperceptible difference between the reference and the test signal [15], [16].

The ODG results for various R-D control schemes discussed in this paper are shown in Fig. 13, in which the reference signal is the original audio sequence. According to the collected test data (Fig. 13), the difference between JTB and CTB schemes is quite small.

The ODG results, comparing against the full search CTB-MNMR scheme, are shown in Fig. 14. Note that the reference signal here is the coded audio sequence produced by the full search CTB-MNMR scheme. We can find that there is almost no difference between the GMNU algorithm and the full search CTB-MNMR particularly at mid to high bit rates. Again the performance of the nonuniform trellis algorithms is better than that of the uniform trellis algorithms with the same computational complexity.

VI. Conclusion

In this paper, we propose a cascaded trellis-based (CTB) optimization scheme to replace the previous joint trellis-based (JTB) scheme for the MPEG-4 AAC coder. The optimization procedure for finding coding parameters, SF and HCB, in the CTB scheme is partitioned into two separated steps. It thus has the advantage of a much reduced computation. The proposed CTB scheme is approximately 71 to 142 times faster than the JTB scheme. Simulation results show that both the objective and subjective quality of the CTB scheme is close to that of the JTB scheme. In addition, we also propose a lossless fast search algorithm for trellis-based optimization on HCB, which provides roughly a four-times speed-up. Furthermore, two nonuniform search algorithms for trellis-based MNMR optimization on SF, so-called GMNU and LMNU, are proposed for reducing calculations. Simulation results indicate that another factor of 25 speed-up can be achieved using GMNU with negligible audio quality loss.

References

[1] T. Painter and A. Spanias, "Perceptual coding of digital audio," *Proc. IEEE*, vol. 88, pp. 451–515, Apr. 2000.
[2] ISO, *Information technology—Coding of Audio-Visual Objects*, 1999. ISO/IEC JTC1/SC29, ISO/IEC IS-14496 (Part 3, Audio).
[3] J. Herre and B. Grill, "Overview of MPEG-4 audio and its applications in mobile communications," in *Proc. WCCC-ICSP*, vol. 1, Aug. 2000, pp. 11–20.
[4] H. Purnhagen, "An overview of MPEG-4 audio version 2," in *Proc. AES 17th Int. Conf. High-Quality Audio Coding*, Firenze, Italy, Sep. 1999.
[5] M. Bosi *et al.*, "ISO/IEC MPEG-2 advanced audio coding," *J. Audio Eng. Soc.*, vol. 45, pp. 789–812, Oct. 1997.
[6] The MPEG Audio Web Page [Online]. Available: http://www.tnt.uni-hannover.de/project/mpeg/audio/.
[7] A. Aggarwal *et al.*, "Trellis-based optimization of MPEG-4 advanced audio coding," in *Proc. IEEE Workshop on Speech Coding*, Sep. 2000, pp. 142–144.
[8] A. Aggarwal *et al.*, "Near-optimal selection of encoding parameters for audio coding," in *Proc. ICASSP*, vol. 5, May 2001, pp. 3269–3272.
[9] G. D. Forney, "The Viterbi algorithm," *Proc. IEEE*, vol. 1, no. 61, pp. 268–278, Mar. 1973.
[10] K. Sayood, *Introduction to Data Compression*, 2nd ed. San Francisco, CA: Morgan Kaufmann, 2000.
[11] S. Golomb, "Run-length encodings," *IEEE Trans. Inform. Theory*, vol. 12, pp. 399–401, Jul. 1966.
[12] H. Najafzadeh and P. Kabal, "Perceptual bit allocation for low rate coding of narrowband audio," in *Proc. ICASSP*, vol. 2, Jun. 2000, pp. 893–896.
[13] EBU, *Sound Quality Assessment Material: Recordings for Subjective Tests*. Brussels, Belgium: Eur. Broadcasting Union, Apr. 1988.
[14] *Method for Subjective Assessment of Small Impairments in Audio Systems Including Multichannel Sound Systems*, 1994. ITU-R BS.1116.
[15] *Method for Objective Measurements of Perceived Audio Quality*, Jul. 2001. Draft ITU-T Recommendation BS.1387.
[16] EAQUAL software, A. Lerchs. [Online]. Available: http://mitiok.free.fr/c.htm.

Cheng-Han Yang was born in 1976. He received the M.Sc. and Ph.D. degrees in electrical engineering from National Chiao Tung University, Hsinchu, Taiwan, R.O.C., in 2000 and 2005, respectively.

Since graduation, he has continued with postdoctoral research at the Electronics Engineering Department, National Chiao Tung University. His research is concentrated in the software/hardware development and DSP implementation of audio coding standards.

Hsueh-Ming Hang received the B.S. and M.S. degrees from National Chiao Tung University, Hsinchu, Taiwan, R.O.C., in 1978 and 1980, respectively, and the Ph.D. degree in electrical engineering from Rensselaer Polytechnic Institute, Troy, NY, in 1984.

From 1984 to 1991, he was with AT&T Bell Laboratories, Holmdel, NJ, and then joined the Electronics Engineering Department, National Chiao Tung University in December 1991. He has been actively involved in international video standards since 1984, and his current research interests include multimedia compression, image/signal processing algorithms and architectures, and multimedia communication systems. He holds seven patents (R.O.C., U.S., and Japan) and has published over 120 technical papers related to image compression, signal processing, and video codec architecture.

Dr. Hang is a recipient of the IEEE Third Millennium Medal and the IEEE ISCE Outstanding Service Award. He was a conference Co-Chair of the Symposium on Visual Communications and Image Processing (VCIP) in 1993, and the Program Chair in 1995. He was a Co-Program Chair for the IEEE International Symposium on Consumer Electronics in 1998 and the IEEE Signal Processing Systems Workshop in 1999. He was an Associate Editor of the IEEE TRANSACTIONS ON IMAGE PROCESSING (1992–1994) and the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY (1997–1999).