

## Enhancement of MPEG-4 ALS Lossless Audio Coding

*Yutaka Kamamoto<sup>†</sup>, Takehiro Moriya, Noboru Harada, and Csaba Kos*

### Abstract

Enhanced codec software that complies with the MPEG-4 Audio Lossless Coding (ALS) has been developed. To improve the compression performance, a new linear prediction analysis method for multichannel coding was devised. Moreover, to reduce the processing time, software implementation has been optimized by utilizing simplified algorithms and efficient instructions for parallel execution. The results of a comprehensive evaluation show that the new software yields a compressed file size that is 0.1% smaller on average and an encoding speed that is as much as six times higher than that of the MPEG reference software. This software will be very useful in practice because the compressed bitstream remains compliant with the international standard.

### 1. Introduction

The MPEG-4 Audio Lossless Coding (ALS) standard was published in March 2006 [1], as a result of official balloting and the efforts of a number of technical contributors, including the Technical University of Berlin (Germany), RealNetworks Corp. (USA), I2R (Singapore), and NTT (Japan) [2]–[6]. This standard is intended for the compression of high-definition audio signals (high sampling rates, high word resolution, and multichannel signals) with the assurance of perfect reconstruction. It provides top-class compression performance and has some additional advantages in terms of interoperability, long-term maintenance, and clear IPR (intellectual property rights) status over other freeware or proprietary technologies. All of these merits have been gained because MPEG-4 ALS is an open international standard. Like other MPEG standards, it specifies the bitstream syntax and the decoding and reconstruction procedures; however, the encoding process has some room for future modification.

In this paper, we describe our efforts to improve the compression performance and reduce the processing time within framework of the MPEG-4 ALS standard.

To improve the compression performance, we have devised a new linear prediction analysis method for multichannel signals. In the conventional multichannel coding (MCC) encoder, adaptively weighted subtraction is carried out between the linear prediction residual signals of the coding channel and that of the reference channel as a result of independent linear prediction. In the new prediction analysis, the subtracted residual signal is taken into account to reduce the code length of the prediction residual signal. In other words, the coefficients of the conventional linear prediction, strictly speaking, do not minimize the subtracted residual signals that can be transmitted, whereas those of our newly devised linear prediction do minimize them.

To reduce the processing time, we investigated the time-consuming components in the ALS encoder, which are the long-term prediction (LTP), MCC, and hierarchical block switching (BS) tools. The existing versions of these tools can significantly improve compression performance, but they do so at the cost of encoder complexity. In our studies, we developed

<sup>†</sup> NTT Communication Science Laboratories  
Atsugi-shi, 243-0198 Japan  
Email: kamamoto.yutaka@lab.ntt.co.jp

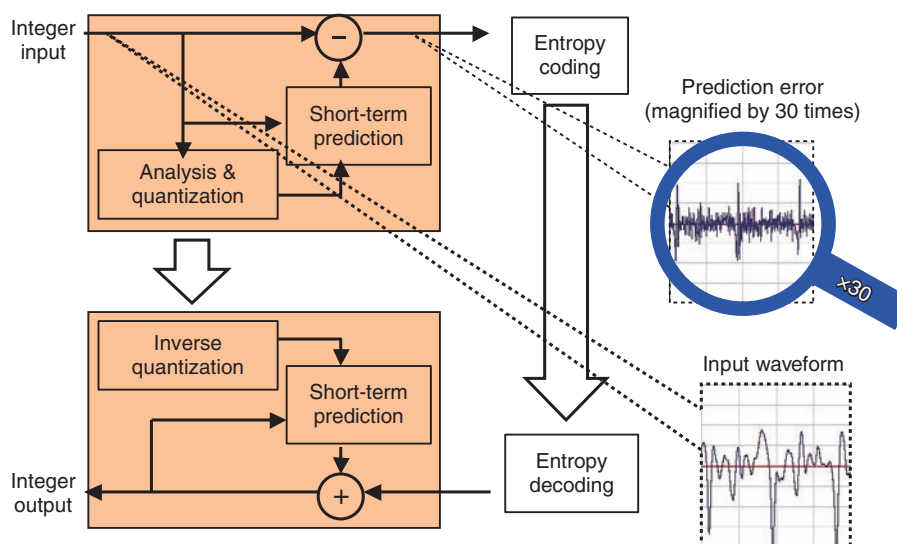


Fig. 1. Fundamental structure of MPEG-4 ALS.

simplified algorithms for LTP, MCC, BS, and other processing modules, starting from the MPEG-4 reference software. In addition, we improved the software implementation and used simultaneous single-input multiple-output execution to enhance the execution speeds of the encoder and decoder.

Several freeware lossless audio codecs are available, with updated versions offered when new features are created or bugs are found. In contrast, an international standard cannot be easily amended, so it is important to make consistent improvement in compression performance, even if by only 0.1%, while maintaining compatibility with the standard. The accumulation of efforts in this area may raise the compression limit.

After briefly overviewing the MPEG-4 ALS structure, we explain our methods for enhancing compression and reducing encoder complexity. Then, we present our experimental results.

## 2. Basic structure of MPEG-4 ALS

### 2.1 Predictive coding

MPEG-4 ALS technology is based on time-domain linear prediction. A diagram outlining the encoding and decoding processes is shown in Fig. 1. Linear prediction technology has been widely used in speech coding systems, such as those used in cellular and IP (Internet protocol) phones. A  $P$ -th order linear prediction analysis provides estimates of the prediction parameters  $a_k (k = 1, 2, \dots, P)$  that minimize the prediction residual  $e(n)$  between the input value  $x(n)$  and

the value  $\hat{x}(n)$  predicted from a given number of past samples, as shown by the following well-known equation ( $a_0 = 1$ ):

$$e(n) = x(n) - \hat{x}(n) = x(n) - \left( -\sum_{k=1}^p a_k \cdot x(n-k) \right) = \sum_{k=0}^p a_k \cdot x(n-k). \quad (1)$$

The integer value of the prediction residual signal and the quantized partial autocorrelation (PARCOR [7]) coefficients obtained from the prediction parameters are transmitted to the decoder. The decoder has a recursive filter that can reconstruct the original waveform losslessly from the transmitted bitstream.

Actually, the residual signal has a smaller amplitude than the input signal does, and the amplitude can be compressed by means of entropy coding using a Golomb-Rice code or block Gilbert-Moore code [8], [9]. Obviously, the smaller the amplitude, the shorter the code length. Therefore, the minimum code length can be obtained approximately by minimizing the energy of the input signal of every frame.

The PARCOR coefficients are also compressed with Rice code and are convenient for checking stability. The prediction order can be adaptively set from 0 (no prediction) to 1023. Additionally, progressive order prediction is used at the initial samples of the random access frames [10].

It should be mentioned that the analysis method for linear prediction is not a concern in the standard bitstream. The Levinson-Durbin (LD) method is implemented in the reference software, though other meth-

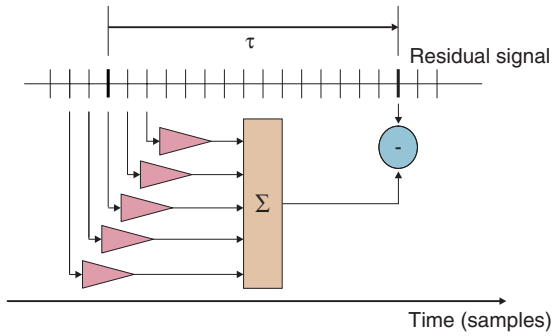


Fig. 2. Structure of five-tap LTP filter with delay parameter  $\tau$ .

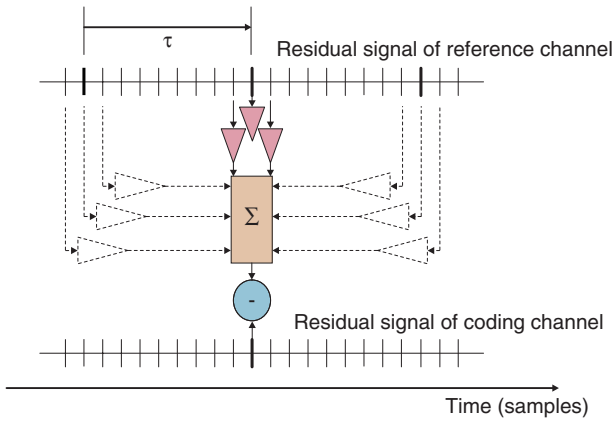


Fig. 3. Structure of MCC filter with relative delay parameter  $\tau$ .

ods, such as the Burg method, the covariance-lattice method [11], and Laguerre-based pure linear prediction (L-PLP) [12], are acceptable for ALS. In contrast, the conventional stereo prediction methods for lossless audio coding [13]–[15] are fundamentally incompatible with those in MPEG-4 ALS in that they need cross-prediction coefficients, whose transmission is forbidden in the ALS bitstream.

### 2.2 LTP

Normal linear prediction, especially the short-term linear prediction described above, utilizes the correlation between neighboring samples to reduce the amplitude. Speech and audio signals sometimes have long-term correlation due to the pitch. This correlation can be used to further reduce the amplitude of the residual signal and thereby reduce the bit rate. As shown in Fig. 2, multi-tap LTP is sequentially applied to the short-term prediction residual signal. To reduce the amplitude, the best delay parameter is found and a set of predictive coefficients is calculated. These are

also compressed by Rice code and transmitted as side information.

At the encoder, short-term prediction residual signal  $e(n)$  is further predicted by the following formula:

$$\tilde{e}(n) = e(n) - \left( \sum_{j=-2}^2 \gamma_j \cdot e(n - \tau + j) \right), \quad (2)$$

where  $\tau$  is the sample lag,  $\gamma_j$  ( $j = -2, -1, \dots, 2$ ) is the quantized gain value,  $e(n)$  is the input sequence, which is a short-term LPC residual sequence, and  $\tilde{e}(n)$  is the subtracted sequence. The most preferable lag  $\tau$  and gain  $\gamma_j$  values are decided in order to reduce the amplitude of the residual signal and these parameters are transmitted as side information. The subtracted samples of this prediction residual signal  $\tilde{e}(n)$  are used for MCC or are directly encoded by the entropy coding.

At the decoder, the reverse process is carried out by means of the recursive filtering

$$e(n) = \tilde{e}(n) + \left( \sum_{j=-2}^2 \gamma_j \cdot e(n - \tau + j) \right), \quad (3)$$

and the reconstructed residual signal  $e(n)$  is used for short-term LPC synthesis.

### 2.3 MCC

There is inter-channel correlation between multiple channels. Inter-channel prediction is applied to the prediction error to reduce the amplitude of the prediction residual after short-term linear prediction or long-term prediction. For multichannel coding, a search is performed to find the channel-pair combination that provides the maximum inter-channel correlation. For the selected channel-pair, multi-tap inter-channel prediction is applied. In addition, the relative delay parameter between the channel-pair is found and the associated weighting coefficients are determined. All these coefficients are then quantized and compressed by Rice code.

Lossless audio coding technology will be widely used for compressing various multichannel signals, such as wave-field-synthesis, bio-medical, and seismic signals as well as surround audio signals. To improve the compression performance for these multichannel signals, adaptive subtraction from reference channels with weighting factors is applied. This process is based on the inter-channel dependence of the time domain prediction residual signal. There are three modes for each channel and each frame: an

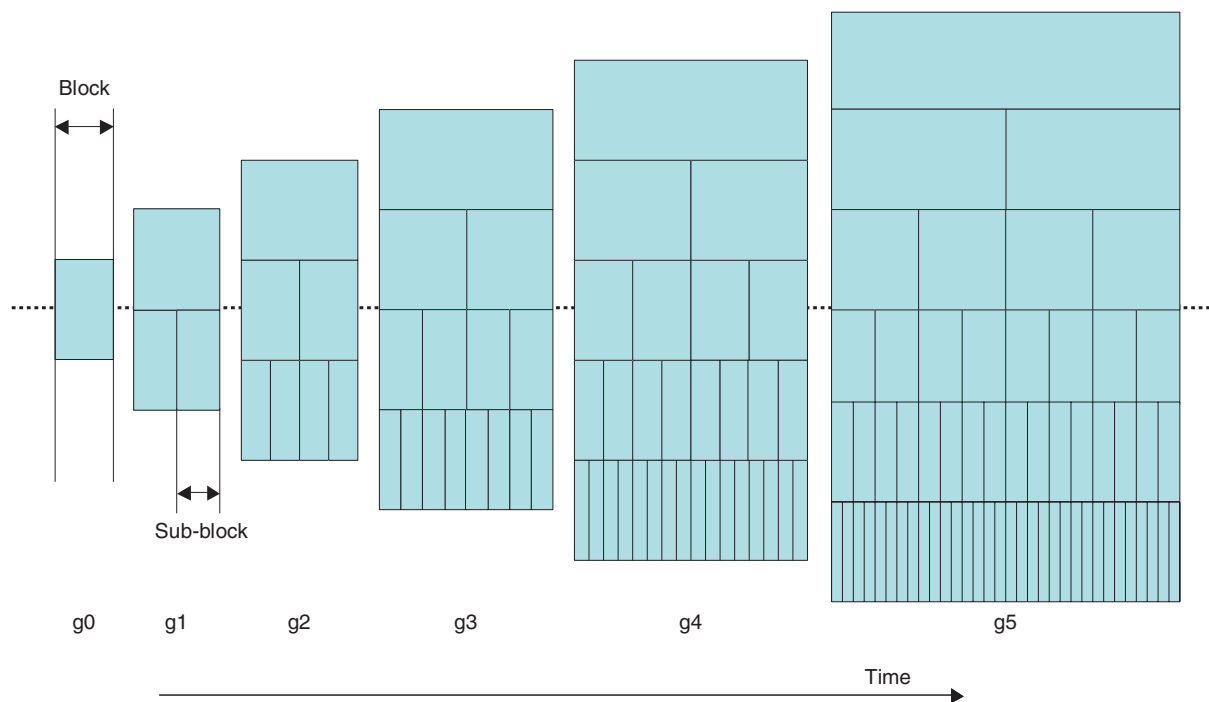


Fig. 4. Hierarchical block switching modes.

independent coding mode, a second mode with three taps, and a third one with six taps (i.e., three delayed taps in addition to the three in the second mode), as shown in **Fig. 3**. At least one channel must be encoded in the independent coding mode for lossless decoding of all the channels.

For the three-tap mode, the operation is performed as follows.

$$\tilde{e}^c(n) = e^c(n) - \left( \sum_{j=-1}^1 \gamma_j \cdot e^r(n+j) \right), \quad (4)$$

where  $\tilde{e}^c(n)$  and  $e^c(n)$  are the residual signals of the coding channel, and  $e^r(n)$  is that of a reference channel. The reference channel is found among available channels and the index is coded together with multi-tap gain parameters  $\gamma_j$  ( $j = -1, 0, 1$ ). The decoder reconstructs the original residual signal by simply applying the reverse operation:

$$e^c(n) = \tilde{e}^c(n) + \left( \sum_{j=-1}^1 \gamma_j \cdot e^r(n+j) \right). \quad (5)$$

The reconstructed residual signal  $e^c(n)$  is used for short-term LPC synthesis or LTP decoding.

For the six-tap mode, adaptive subtraction is carried out using

$$\tilde{e}^c(n) = e^c(n) - \left( \sum_{j=-1}^1 \gamma_j \cdot e^r(n+j) + \sum_{j=-1}^1 \gamma_{j+\tau} \cdot e^r(n+\tau+j) \right), \quad (6)$$

where a lag parameter  $\tau$  can be estimated by cross correlation between the coding channel and the reference channel, and multi-tap gain parameters  $\gamma_j$  ( $j = -1, 0, 1, \tau-1, \tau, \tau+1$ ) can be obtained by minimizing the energy of the subtracted residual sequence. In the decoder, the original residual signal is inversely reconstructed as follows:

$$e^c(n) = \tilde{e}^c(n) + \left( \sum_{j=-1}^1 \gamma_j \cdot e^r(n+j) + \sum_{j=-1}^1 \gamma_{j+\tau} \cdot e^r(n+\tau+j) \right). \quad (7)$$

Again, the reconstructed residual signal  $e^c(n)$  is used for short-term LPC synthesis or LTP decoding.

## 2.4 Hierarchical BS

The ALS standard has a hierarchical block-switching scheme that further enhances compression performance, as shown in **Fig. 4**. The encoder can select the best combination of sub-blocks in a frame. Mode g1 can adaptively select two sub-block lengths for every

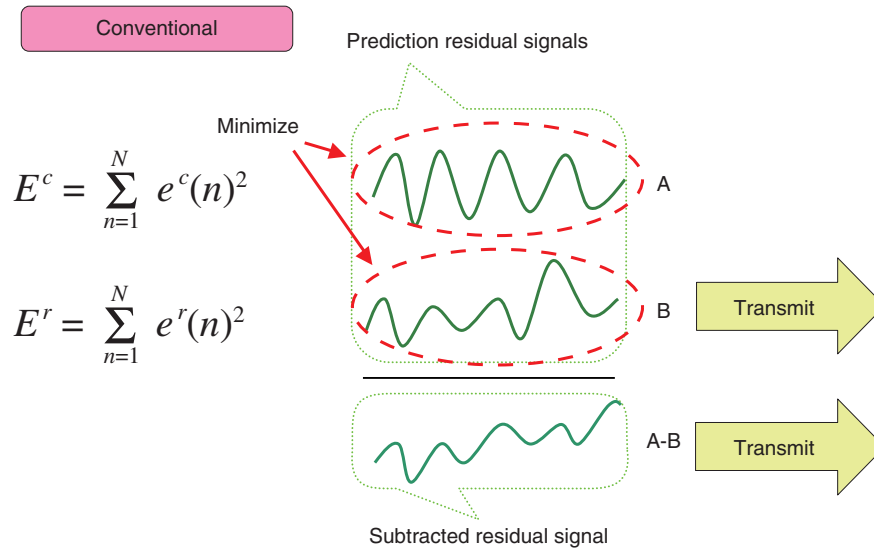


Fig. 5. Conceptual diagram of conventional method.

block, while mode g0 can use only one block length. The number of sub-blocks can be 32 in the case of mode g5. A longer block length tends to be useful for stationary signals, while a shorter block length is effective for non-stationary signals.

### 2.5 Other features

For a floating-point signal, MPEG-4 ALS has a compression tool that makes use of the fundamental compression structure of an integer signal [16], [17]. The floating-point format is useful for professional mixing of music because there is no risk of overflow or underflow. However, this format cannot be compressed because its nominal value has no correlation between samples. We have therefore invented a novel scheme that decomposes a floating-point signal sequence into integer and remaining sequences. We have also introduced the approximate common factor scheme, which greatly improves the compression performance when an input floating-point value sequence is generated from the integer value sequence multiplied by a common number throughout the frame. The common number can be detected by means of rational approximations even though the input samples have errors due to truncation or some other operations. The remaining sequence is further compressed by masked Lempel-Ziv compression, making full use of the properties inherited from the decomposition processes.

There are a number of other control parameters for the encoder, and it should be possible to efficiently estimate them in the future without losing conformity

to the standard [18], [19].

## 3. Enhancement of MPEG-4 ALS

### 3.1 Linear prediction tool for MCC

In conventional linear prediction analysis methods, as shown in Fig. 5, the prediction parameters are estimated to minimize the prediction residuals of each channel independently. Therefore, the prediction parameters of the reference channel  $a_k^r (k = 1, 2, \dots, P^r)$  and those of the coding channel  $a_k^c (k = 1, 2, \dots, P^c)$  minimize energy  $E^r = \sum_{n=1}^N (e^r(n))^2$  and energy  $E^c = \sum_{n=1}^N (e^c(n))^2$ , where  $N$  denotes the number of samples per frame.

However, in the coding channel, we should minimize the energy of the subtracted signal

$$\tilde{E}^c = \sum_{n=1}^N (\tilde{e}^c(n))^2$$

when we use MCC because, as shown in Fig. 6, the residual  $e^c(n)$  is not transmitted but the subtracted residual  $\tilde{e}^c(n)$  is. Consequently, we obtain the prediction parameters of the coding channel that minimizes  $\tilde{E}^c$  and thereby improve the compression performance. To simplify the explanation without losing generality, we assume one-tap subtraction:

$$\tilde{e}^c(n) = e^c(n) - \gamma \cdot e^r(n), \tag{8}$$

where, to minimize  $\tilde{E}^c$ , we calculate  $\gamma$  as

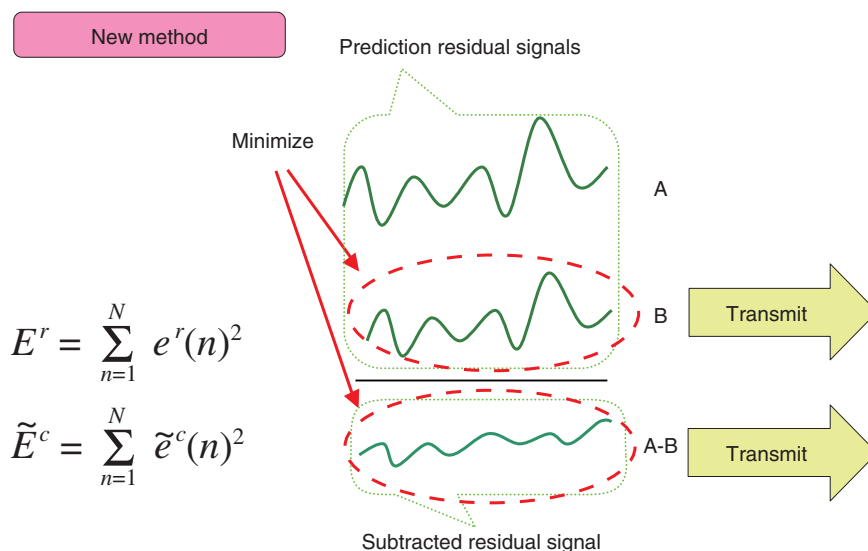


Fig. 6. Conceptual diagram of our new method.

$$\gamma = \frac{\sum_{n=1}^N e^c(n) \cdot e^r(n)}{\sum_{n=1}^N e^r(n) \cdot e^r(n)}. \quad (9)$$

From Eqs. (6) and (7),  $\tilde{E}^c$  can be rewritten as ( $a_0^c = 1$ ):

$$\begin{aligned} \tilde{E}^c &= \sum_{n=1}^N (\tilde{e}^c(n))^2 = \sum_{n=1}^N (e^c(n) - \gamma \cdot e^r(n))^2 \\ &= \sum_{n=1}^N \left( \sum_{k=0}^{P^c} a_k^c \cdot x^c(n-k) \right)^2 \\ &\quad - \frac{\left( \sum_{n=1}^N e^r(n) \cdot \sum_{k=0}^{P^c} a_k^c \cdot x^c(n-k) \right)^2}{E^r} \end{aligned} \quad (10)$$

Then, to minimize Eq. (10), we derive the prediction parameters  $a_k^c$  ( $k = 1, 2, \dots, P^c$ ) from the partial differential

$$\frac{\partial \tilde{E}^c}{\partial a_i^c} = 0 \quad (i = 1, 2, \dots, P^c) \quad (11)$$

by solving the normal equation

$$\mathbf{a} = \mathbf{V}^{-1} \cdot \mathbf{w}, \quad (12)$$

where

$$\mathbf{a} = (a_1^c, a_2^c, \dots, a_{P^c}^c)^T$$

$$\mathbf{V} = \begin{pmatrix} R(0) - \frac{C(1,1)}{E^r} & \dots & R(P^c-1) - \frac{C(1,P^c)}{E^r} \\ \vdots & \ddots & \vdots \\ R(P^c-1) - \frac{C(1,P^c)}{E^r} & \dots & R(0) - \frac{C(P^c,P^c)}{E^r} \end{pmatrix}$$

$$\mathbf{w} = \begin{pmatrix} R(1) - \frac{C(0,1)}{E^r} \\ \vdots \\ R(P^c) - \frac{C(0,P^c)}{E^r} \end{pmatrix}$$

$$R(\tau) = \sum_{n=1}^N x^c(n) \cdot x^c(n-\tau)$$

$$C(\mu, \nu) = \left( \sum_{n=1}^N e^r(n) \cdot x^c(n-\mu) \right) \cdot \left( \sum_{n=1}^N e^r(n) \cdot x^c(n-\nu) \right).$$

Equation (12) can be solved by the Cholesky decomposition method because matrix  $\mathbf{V}$  is not a Toeplitz matrix but a symmetric one. Sometimes a synthesis filter with these prediction parameters may be unstable. In such a case, the usual Levinson-Durbin method can be used to minimize  $E^c$  instead of our new method. In addition, the bitstreams derived from the conventional method and from our new method are compared and the smaller one is transmitted. This guarantees that the best possible compress-

sion performance is always obtained.

### 3.2 Complexity reduction

#### 3.2.1 Calculating auto-correlation coefficients

Auto-correlation coefficients are essential for short-term linear prediction analysis. The computational cost becomes greater as the prediction order and/or the number of samples per frame increases. To reduce the computational complexity, fast Fourier transformation (FFT) is used when the prediction order is large. Similarly, the most time-consuming operation in the LTP encoder is the auto-correlation calculation for determining delay parameter  $\tau$ . For sampling rates of 192 kHz or higher, delay parameter values can range from the maximum short-term prediction order to that plus 1024. Under these conditions, FFT can significantly reduce the processing time for calculating auto-correlation coefficients.

#### 3.2.2 Simplified algorithm for MCC

MCC encoding and decoding processes are flexibly defined for signals with various numbers of channels up to 65,536. Selecting the reference channel generally requires a computational load. For music applications, two-channel stereo signals are the most important. As far as stereo input is concerned, the procedure for selecting the reference channel can be greatly reduced by simply comparing the energy of the two signals. In addition, only a three-tap filter can be used for stereo signals, although a six-tap filter is effective for multichannel signals recorded using a linear array of microphones.

#### 3.2.3 Simplified hierarchical BS

To find the best sub-block division, we need to calculate the actual code length for all sub-block combinations exhaustively. The total processing time is roughly proportional to the number of layers in the hierarchy. We can reduce the complexity by means of a sub-optimal search using the estimated code length and by reusing the intermediate calculation results among layers.

## 4. Experimental evaluation

### 4.1 Conditions

To evaluate our methods, we performed encoding and decoding experiments for musical signals. As input signals, we used 13 copyright-free music items of CD (compact disc) quality (stereo, sampling rate: 44.1 kHz, word length: 16 bits, duration: 64 minutes, file size: 642 MB) composed by one of the authors

(Y.K.) and 20 CD songs with English lyrics from the RWC music database (RWC-MDB-P-2001 No. 81–100, 85 min., 855 MB) [20]. These are similar to copyright popular songs. The obtained encoding or decoding times are the total processing times for all input signals (the faster the better) measured in seconds using “timeit.exe” on a Microsoft Windows Server 2003 with an AMD Opteron processor operating at 2.4 GHz with 2 GB of memory and normalized by the duration time (the smaller the faster; 50% means double speed). We defined the compression ratio (the smaller the better) in percent as

$$\begin{aligned} \text{compression ratio (\%)} \\ = \frac{\text{compressed file size}}{\text{original file size}} \times 100. \end{aligned} \quad (13)$$

### 4.2 Results

The experimental results are shown in Figs. 7–12, where the six symbols on each curve correspond to systems with different maximum prediction orders of 7, 15, 31, 63, 127, and 255, from top to bottom. The encoding time and compression ratio with MCC are also shown in Fig. 7. Our software (red line: NTT with MCC) reduced the encoding time to 1/6 compared with the reference software (blue line: RM with MCC). Furthermore, our linear prediction tool for MCC (purple line: NTT with MCC and new predictor) yielded 0.1% better compression than the conventional MCC.

At the decoder, as shown in Fig. 8, the decoding time was reduced to 1/2 of the reference model (blue line: RM with MCC) as a result of the efficient software implementation (red line: NTT with MCC). And the new prediction tool (purple line: NTT with MCC and new predictor) offered better compression performance than the conventional one (red line) with no increase in decoding time. In other words, the bit-stream compressed by the developed method can be decoded faster than the one compressed by the conventional method with higher prediction order, yielding the same compression performance.

Encoding times for LTP are compared in Fig. 9. The encoding time could be reduced to 1/5 of the reference model (blue line: RM with LTP) by the simplified encoder (red line: NTT with LTP). The decoding time and compression ratio with LTP are shown in Fig. 10. The decoding time was reduced to 1/2 that of the reference model (blue line: RM with LTP) because of the optimized implementation (red line: NTT with LTP).

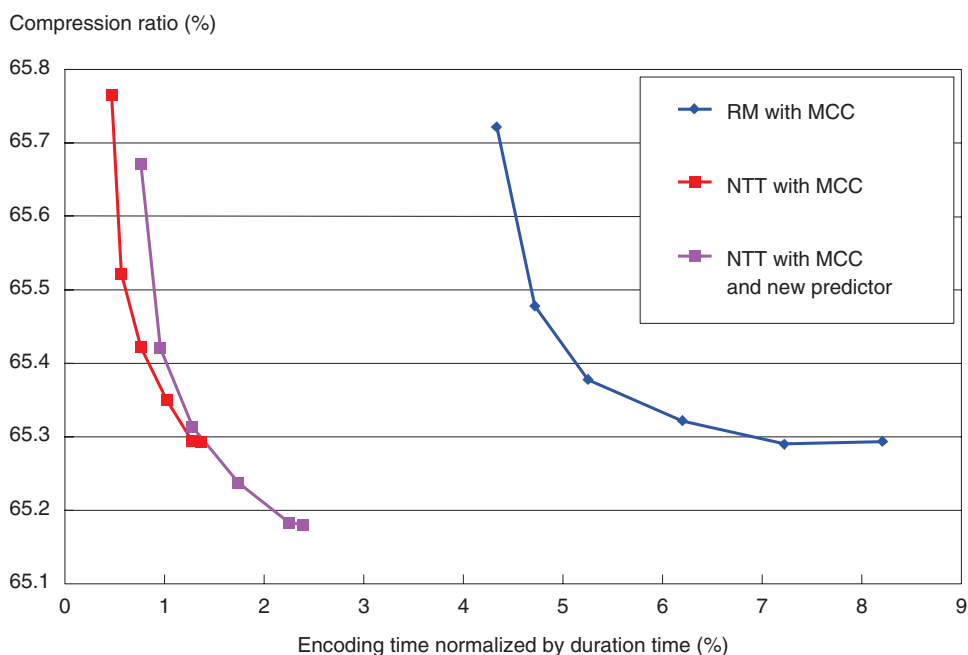


Fig. 7. Encoding times and compression ratios with MCC (left is faster; lower is smaller).

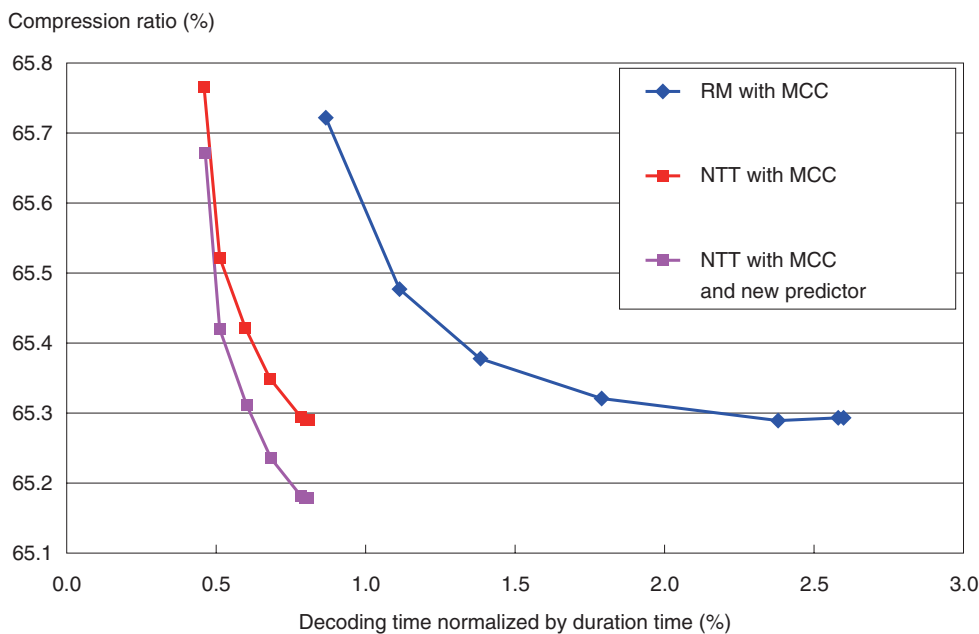


Fig. 8. Decoding times and compression ratios with MCC (lower left is better).

The BS encoding time and compression ratio are shown in Fig. 11. Our software (purple and orange lines: NTT with BS levels 1 and 2) reduced the encoding time to 1/3 compared to the reference model (light blue and green lines: RM with BS levels

1 and 2). The difference in processing speed became larger as the block switching level increased. Decoding times for BS are shown in Fig. 12. We can see that the software improved by NTT (red, purple, and orange lines: NTT) could reduce the decoding time



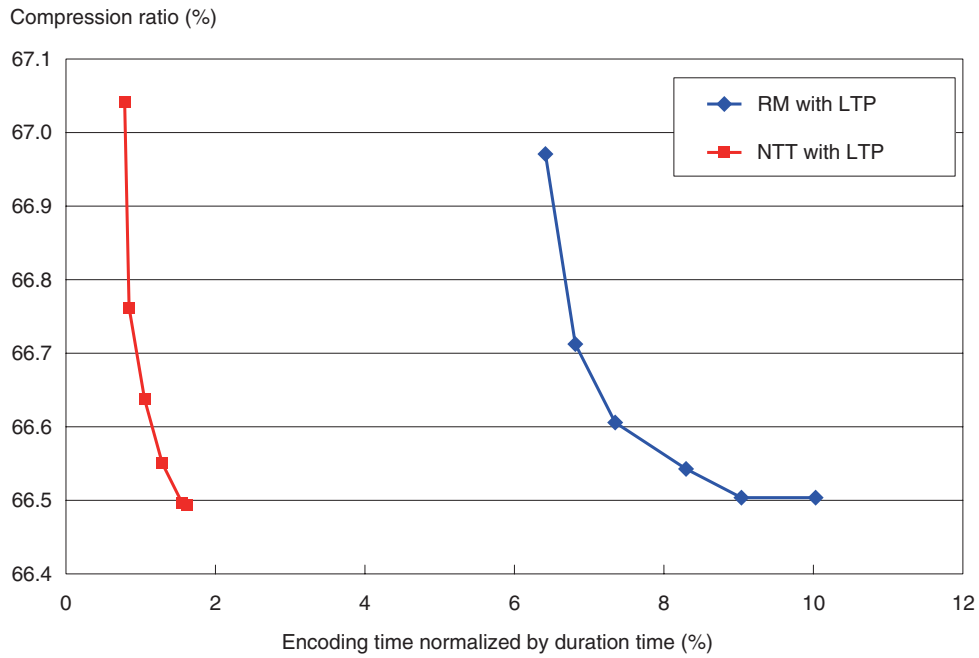


Fig. 9. Encoding times and compression ratios with LTP (left is faster; lower is smaller).

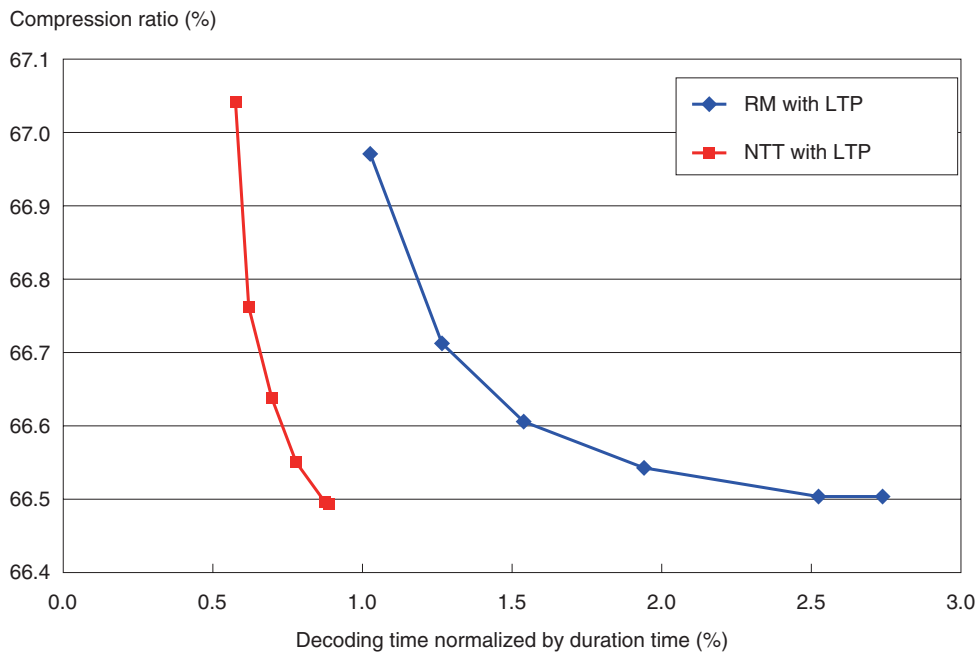


Fig. 10. Decoding times and compression ratios with LTP (lower left is better).

by 1/3, compared with the reference model (blue, light blue, and green lines: RM).

In summary, the software enhanced by NTT can generate a smaller bitstream, while remaining com-

pliant to MPEG-4 ALS. As shown in Figs. 7–12, the processing time of this enhanced software (NTT proprietary) is shorter than that of the reference model software (RM) overall.

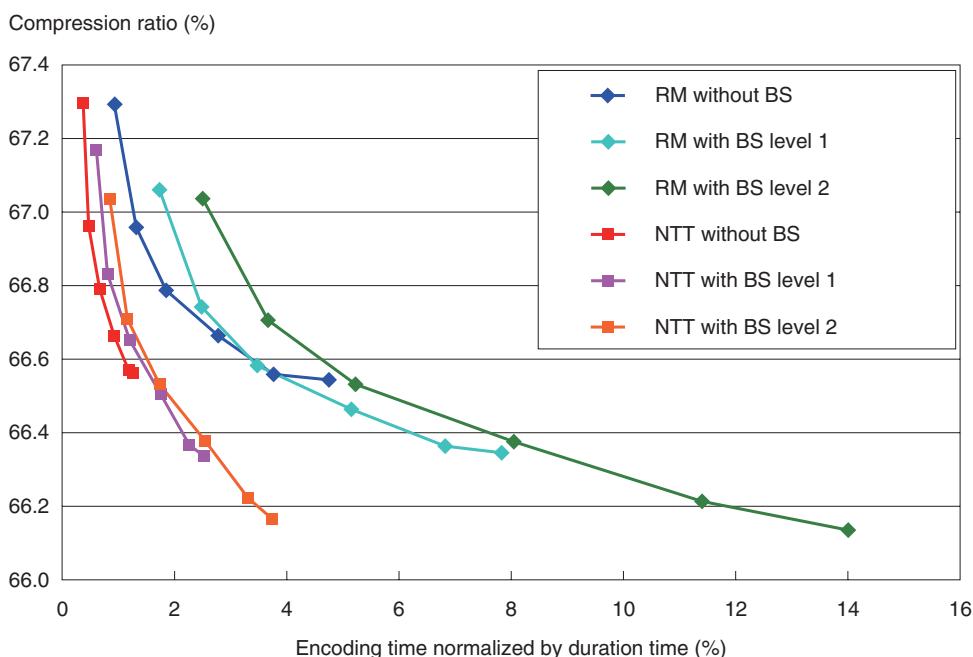


Fig. 11. Encoding times and compression ratios with BS (left is faster; lower is smaller).

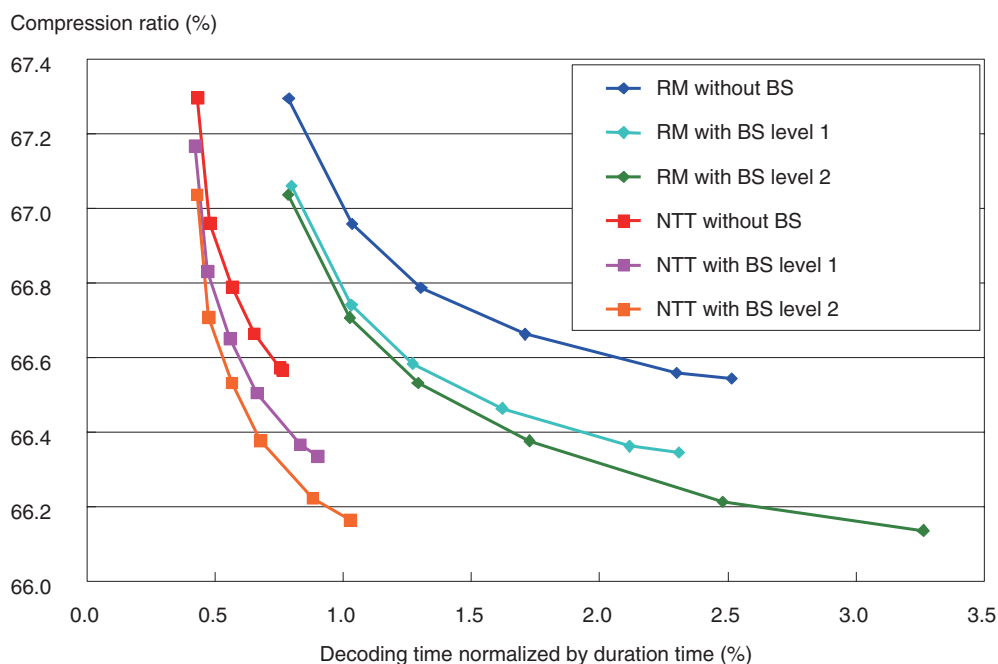


Fig. 12. Decoding times and compression ratios with BS (lower left is better).

### 5. Conclusion

We have developed enhanced codec software that complies with the MPEG-4 Audio Lossless Coding

(ALS). A comprehensive evaluation showed that it yields a slightly smaller (0.1%) compressed file size on average and an encoding speed that is as much as six times higher than that of the MPEG reference

software. For the decoder, efficient software implementation reduced the processing time by half. This software is very practical because the compressed bitstream remains compliant with the international standard. This standard is expected to provide common tools for various applications [21] and should continue to be maintained so that the compressed files can be perfectly decoded even 100 years from now. Potential applications include audio signal archiving, professional audio editing, portable music players, and editing or archiving time series signals other than audio ones, such as medical or environmental data.

## References

- [1] ISO/IEC 14496-3:2005/Amd.2:2006, Information technology—Coding of audio-visual objects—Part 3: Audio, Amendment 2: Audio Lossless Coding (ALS), new audio profiles and BSAC extensions, edition 2006-03-15.
- [2] T. Liebchen, T. Moriya, N. Harada, Y. Kamamoto, and Y. A. Reznik, “The MPEG-4 audio lossless coding (ALS) standard technology and applications,” in Preprint AES 119th Convention #6589, NY, USA, 2005.
- [3] T. Moriya, N. Harada, Y. Kamamoto, and H. Sekigawa, “MPEG-4 ALS—International Standard for Lossless Audio Coding,” NTT Technical Review, Vol. 4, No. 8, pp. 40–45, 2006.
- [4] Y. Kamamoto, T. Moriya, T. Nishimoto, and S. Sagayama, “Lossless compression of multi-channel signals based on inter-channel correlation,” *IPSSJ Trans.*, Vol. 46, No. 5, pp. 1118–1128, 2005 (in Japanese).
- [5] Y. Kamamoto, T. Moriya, N. Harada, T. Nishimoto, and S. Sagayama, “Intra- and Inter-Channel Long-Term Prediction in ISO/IEC MPEG-4 Audio Lossless Coding (ALS),” *IEICE Trans. on Communications*, Vol. J89-B, No. 2, pp. 214–222, 2006 (in Japanese).
- [6] N. Harada, T. Moriya, H. Sekigawa, K. Shirayanagi, and Y. Kamamoto, “Lossless Compression of IEEE754 Floating-point Signal in ISO/IEC MPEG-4 Audio Lossless Coding (ALS),” *IEICE Trans. on Communications*, Vol. J89-B, No. 2, pp. 204–213, 2006 (in Japanese).
- [7] J. Makhoul, “Linear prediction: A tutorial review,” *Proc. of IEEE*, Vol. 63, No. 4, pp. 561–580, 1975.
- [8] T. Liebchen and Y. A. Reznik, “MPEG-4 ALS: an emerging standard for lossless audio coding,” *Proc. of Data Compression Conference (DCC) 2004*, pp. 439–448, Utah, USA, 2004.
- [9] Y. A. Reznik, “Coding of prediction residual in MPEG-4 standard for lossless audio coding (MPEG-4 ALS),” *Proc. of ICASSP 2004*, pp. III-1024–1027, 2004.
- [10] T. Moriya, D. Yang, and T. Liebchen, “Extended Linear Prediction Tools for Lossless Audio Coding,” *Proc. of ICASSP 2004*, pp. III-1008–1011, 2004.
- [11] J. Makhoul, “Stable and efficient lattice methods for linear prediction,” *IEEE Trans. Acoustics, Speech and Signal Processing*, Vol. 25, No. 5, pp. 423–428, 1977.
- [12] A. Biswas and B. den Brinker, “Lossless compression of digital audio using Laguerre-based pure linear prediction,” *Proc. SPS 2004 (4th IEEE Benelux Signal Processing Symp.)*, pp. 49–52, 2004.
- [13] T. Liebchen, “Lossless audio coding using adaptive multichannel prediction,” in Preprint AES 113rd Convention, #5680, LA, USA, 2002.
- [14] A. Biswas and A. C. den Brinker, “Quantization of transmission parameters in stereo linear prediction system,” *Proc. of Data Compression Conference (DCC) 2006*, pp. 262–271, 2006.
- [15] F. Ghido and I. Tabus, “Adaptive design of the preprocessing stage for stereo lossless audio compression,” in Preprint AES 122nd Convention, #7085, Vienna, Austria, 2007.
- [16] D. Yang and T. Moriya, “Lossless compression for audio sources with IEEE floating-point format,” in Preprint AES 115th Convention, #5987, NY, USA, 2003.
- [17] N. Harada, T. Moriya, H. Sekigawa, and K. Shirayanagi, “Lossless compression of IEEE floating-point audio using approximate common factor coding and masked-LZ compression,” in Preprint AES 118th Convention, #6352, Barcelona, Spain, 2005.
- [18] T. Moriya, N. Harada, and Y. Kamamoto, “An enhanced encoder for the MPEG-4 ALS lossless coding standard,” in Preprint AES 121st Convention, # 6869, San Francisco, USA, 2006.
- [19] T. Moriya, N. Harada, and Y. Kamamoto, “Performance-complexity tradeoffs of the MPEG-4 ALS lossless coding standard,” *Proc. of IEEE 40th Asilomar Conference on Signals, Systems and Computers*, WA7a-4, pp. 2130–2134, 2006.
- [20] M. Goto, “Development of the RWC music database,” *Proc. of ICA2004 (18th International Congress on Acoustics)*, pp. I-553–556, 2004.
- [21] N. Harada, T. Moriya, and Y. Kamamoto, “An audio archiving format based on MPEG-4 Audio Lossless Coding,” in Preprint AES 121st Convention, #6895, San Francisco, USA, 2006.


**Yutaka Kamamoto**

Researcher, Moriya Research Laboratory, NTT Communication Science Laboratories.

He received the B.S. degree in applied physics from Keio University, Kanagawa, in 2003 and the M.S. degree in information physics and computing from the University of Tokyo, Tokyo, in 2005. Since joining NTT Communication Science Laboratories in 2005, he has been studying signal processing and information theory. He is a member of the Acoustical Society of Japan (ASJ), the Information Processing Society of Japan (IPSJ), the Society of Information Theory and its Applications, the Institute of Electronics, Information and Communication Engineers (IEICE) of Japan, and IEEE.


**Noboru Harada**

Research Scientist, Moriya Research Laboratory, NTT Communication Science Laboratories.

He received the B.S. and M.S. degrees from the Department of Computer Science and Systems Engineering of Kyushu Institute of Technology, Fukuoka, in 1995 and 1997, respectively. He joined NTT Human Interface Laboratories in 1997. His main research area has been lossless audio coding and high-efficiency coding of speech and audio. He is a member of ASJ, IEICE, the Audio Engineering Society, and IEEE.


**Takehiro Moriya**

Research Fellow, Moriya Research Laboratory, NTT Communication Science Laboratories.

He received the B.S., M.S., and Ph.D. degrees all in applied mathematics and instrumentation physics from the University of Tokyo, Tokyo, in 1978, 1980, and 1989, respectively. Since joining the Musashino Electrical Communication Laboratories of Nippon Telegraph and Telephone Public Corporation (now NTT) in 1980, he has been engaged in research on and the standardization of speech and audio coding. In 1989, he stayed at AT&T Bell Laboratories as a guest researcher. He is a member of ASJ, IPSJ, and IEICE and a fellow of IEEE.


**Csaba Kos**

“Vulcanus in Japan” intern, Moriya Research Laboratory, NTT Communication Science Laboratories.

He received the M.Sc. degree in computer science from Eötvös Loránd University (ELTE, Hungary) in 2006. He wrote his MSc thesis at the Hungarian Academy of Sciences, Computer and Automation Research Institute (MTA SZTAKI) in the area of three-dimensional scene reconstruction and camera autocalibration. In 2006, after six months as a product developer at Nokia, he was awarded the “Vulcanus in Japan” grant. As part of this, he has spent eight months at NTT Communication Science Laboratories. Besides audio coding, his current interests include computer vision, mobile telecommunications, and rapidly developing topics such as GPGPU programming and parallel programming.