Low-complexity PARCOR coefficient quantization and prediction order estimation designed for entropy coding of prediction residuals

Yutaka Kamamoto1,∗, Takehiro Moriya1,†, Noboru Harada1,‡ and Yusuke Hiwasaki2,§
1NTT Communication Science Laboratories, Nippon Telegraph and Telephone Corporation, 3–1 Morinosato-Wakamiya, Atsugi, 243–0198 Japan
2NTT Media Intelligence Laboratories, Nippon Telegraph and Telephone Corporation, 3–9–11 Midori-cho, Musashino, 180–8585 Japan
(Received 28 May 2012, Accepted for publication 23 October 2012)

Abstract: This paper presents a set of low-complexity tools used in lossless coding of G.711 bitstream, based on linear prediction. One is an algorithm for quantizing the PARCOR/reflection coefficients and the other is an estimation method for the optimal prediction order. Both tools are based on a criterion that minimizes the entropy of the prediction residual signal and can be implemented in fixed-point arithmetic at very low-complexity. Since proposed methods show efficient performance in terms of compression and complexity, they are adopted in the Recommendation ITU-T G.711.0, a new standard for lossless compression of G.711 (A-law/μ-law logarithmic PCM) payload.

Keywords: Speech coding, Lossless compression, Linear predictive coding, Standardization, ITU-T G.711.0

PACS number: 43.72.Gy [doi:10.1250/ast.34.105]

1. INTRODUCTION

Recommendation ITU-T G.711 is the benchmark speech coding technology for narrow-band telephony applications [1]. Recently, it is also used for packet-based networks such as voice over Internet Protocol (VoIP). In order to reduce the bit rate of the G.711 payload without any degradation, a new ITU-T standard G.711.0 was established [2]. In other words, G.711.0 enables the compression of the A-law/μ-law logarithmic pulse-code modulation (PCM) signal losslessly.

Since the G.711 is used not only for sending speech signal but also for transmitting facsimile and modem data, the G.711.0 has several efficient tools to encode the signal that is not fit into a source-filter model [3]. In fact, the common use case is voice conversation, thus most input signal is processed by mapped-domain linear predictive coding (MDLPC). In order to avoid the error propagation caused by packet-loss, the G.711.0 does not exploit inter-frame redundancies. All process is completed within a frame, of which length corresponds 5 ms, 10 ms, 20 ms, 30 ms and 40 ms that are typically used in IP networks.

G.711.0 has been designed to have low complexity (1.0 WMOPS in average; less than 1.7 WMOPS for the worst case) and low memory footprint (less than 5k octets RAM, 5k octets ROM, and 3.6k basic operations). These features are extremely useful for compressing IP streams on gateway routers that handle multiple links [4]. The G.711.0 codec has been characterized to provide over 50% average compression in typical service provider environments. Overall compression is primarily a function of the signal type and level, background noise level, and encoding law (A-law/μ-law).

Simply speaking, this new compression technology is based on linear prediction and entropy coding; therefore, the quantization method of the partial autocorrelation (PARCOR) coefficients that will be quantized and transmitted affects the compression performance, especially for shorter frame size such as 40 or 80 samples per frame in 8kHz sampling. For lossy coding, several quantization algorithms of PARCOR coefficients, also known as reflection coefficients, have been extensively investigated [5–8]. However, with G.711.0, there is a need for such a quantization algorithm for lossless coding. In general speech coding schemes for IP and mobile phones such as ITU-T G.729 and 3GPP adaptive multi-rate (AMR), line spectrum pair (LSP) is usually used to compress the linear prediction coefficients. This parameter has significant
merits of compression with interpolation, vector quantization and prediction. However, a conversion and a quantization consume typically about 1.0 WMOPS [9,10]. Thus, LSP parameter is not feasible for this application.

In this paper, two novel compression tools are described: a low-complexity algorithm to quantize PARCOR coefficients and a low-complexity prediction-order estimation scheme for lossless coding. These two approaches are based on a criterion that minimizes the entropy of the prediction residual signal [11–14], while the computation complexity is kept significantly small. These tools show efficient performance in terms of compression and complexity, and they are included in the G.711.0 standard.

The remainder of this paper is organized as follows. Section 2 introduces the basic structure of MDLPC. Then, we describe a low-complexity PARCOR coder for entropy coding of prediction residuals in Sect. 3 and a low-complexity prediction-order estimator in Sect. 4. Finally, Sect. 5 gives the conclusion.

2. BASIC STRUCTURE OF MAPPED DOMAIN LINEAR PREDICTIVE CODING IN G.711.0

As described in the previous section, most of the frames are encoded by MDLPC. The coding tool takes a sequence of \( N \), number of samples per frame, G.711 A-law symbols \( x_A(n) \) \( (n = 1, 2, \cdots, N) \) or G.711 \( \mu \)-law symbols \( x_\mu(n) \). First, these G.711 symbols are converted into \( x_{pcm}(n) \) in the uniform PCM domain (also known as linear PCM domain) by using an A-law/\( \mu \)-law to uniform PCM conversion (i.e., decoding into the uniform PCM domain) as follows

\[
x_{pcm}(n) = \begin{cases} 
  f_{A\rightarrow pcm}(x_A(n)) & \text{for A-law} \\
  f_{\mu\rightarrow pcm}(x_\mu(n)) & \text{for } \mu\text{-law}
\end{cases}
\]  

(1)

and the residual signal that are encoded by Golomb-Rice and the Levinson-Durbin algorithm, and then quantized by the method described in Sect. 3.

where \( f_{pcm\rightarrow int8} \) denotes the function for converting from the uniform PCM to the \( int8 \) domain as described in Fig. 2. The range of residual signal \( e(n) \) is limited in the range of \(-255 \) to \(255\). Prediction parameters \( \hat{a}_i \) \((i = 1, 2, \cdots, P)\) are derived from quantized PARCOR coefficients, which are first obtained by the Levinson-Durbin algorithm, and then quantized by the method described in Sect. 3.

By transmitting the quantized PARCOR coefficients and the residual signal that are encoded by Golomb-Rice code [15–18], adaptive recursive Golomb-Rice code or Huffman code with escape symbol (E-Huffman) [19,20], a recursive filter at the decoder can produce original waveform \( \hat{x}_{int8}(n) \) losslessly. Consequently, the original G.711 data \( x_A/\mu(n) \) are also reconstructed perfectly. This mapping strategy has also been discussed in other lossless compression schemes [21,22].

3. LOW-COMPLEXITY PARCOR QUANTIZATION FOR ENTROPY CODING OF PREDICTION RESIDUALS

3.1. Estimation of Approximate Code Length

Owing to \( int8/pcm \) conversion, MDLPC is approxi-
mated as normal LPC. Here, we assume that mean of input speech signal is zero and that Golomb-Rice code [15–18], adaptive recursive Golomb-Rice code, and E-Huffman code [19,20] asymptotically approach the entropy of residual signal when the distribution of prediction residuals follows Gaussian distribution or Laplace distribution. The prediction order P is appropriately selected by means of characteristics of input signal (e.g., P = 0 or P = 1 for white noise input). As used in the intermediate process of the Levinson-Durbin algorithm, the energy of residual signal after P-th order prediction, E(P), can be estimated by PARCOR coefficients as follows

\[ E(P) = \sum_{n=1}^{N} (e_p(n))^2 = \sum_{n=1}^{N} \left( \sum_{i=0}^{P} a_i \cdot x(n-i) \right)^2 = E(0) \cdot \prod_{i=1}^{P} (1 - k_i^2), \]

where, \( E(0) = \sum_{n=1}^{N} (x(n))^2 \) is an energy of input signal, \( a_i \) (i = 1, 2, ..., P) are prediction coefficients (\( a_0 = 1 \)), and \( k_i \) (i = 1, 2, ..., P) are the PARCOR coefficients (\( \|k_i\| < 1 \)). So, the variance of P-th order residual signal becomes

\[ \sigma^2 = \frac{E(P)}{N} = \frac{E(0)}{N} \cdot \prod_{i=1}^{P} (1 - k_i^2). \]

When the signal follows a Gaussian distribution, their entropy can be denoted as

\[ H_G(\sigma^2_p) = \log_2(\sqrt{2\pi e\sigma^2_p}), \]

and when follow Laplace distribution, the entropy is described as

\[ H_L(\sigma^2_p) = \log_2(\sqrt{2e^2\sigma^2_p}). \]

Since these entropies depend on the variance \( \sigma^2_p \), Eqs. (6) and (7) can be transcribed as

\[ H(\sigma^2_p) = \beta + \frac{1}{2} \log_2(\sigma^2_p), \]

where the constant \( \beta \) is about 2.0 (\( \approx \log_2(\sqrt{2\pi e}) \)) and 1.9 (\( \approx \log_2(\sqrt{2e^2}) \)) for Gaussian and Laplace distributions, respectively. From Eqs. (5) and (8), the estimated code length of prediction residuals per frame is

\[ C_e(P) = H(\sigma^2_p) \cdot N = \left\{ \beta + \frac{1}{2} \log_2(\sigma^2_p) \right\} \cdot N \]

\[ = \frac{N}{2} \left\{ 2\beta + \log_2 \left( \frac{E(0)}{N} \right) \right\} + \sum_{i=1}^{P} \log_2(1 - k_i^2). \]

The first term of this equation means that the code length depends on the statistical distribution of prediction residual signal. The second and third terms denote that the characteristic of input signal affects the compression performance. When the amplitude of input speech is large, the code length becomes long. Even though the energy of input signal is high, we may have shorter code length if the signal has redundancy for time to time because the third term becomes big negative value.

To obtain the total code length per frame, additional bits \( C_{\gamma}(P) = \sum_{i=1}^{P} \gamma_i \) that are the bits needed to represent the all PARCOR coefficients corresponding to \( k_i \) (i = 1, 2, ..., P). This means that the total code length per frame is described as

\[ C(P) = C_e(P) + C_{\gamma}(P) = C_e(P) + \sum_{i=1}^{P} \gamma_i \]

\[ = \frac{N}{2} \left\{ 2\beta + \log_2 \left( \frac{E(0)}{N} \right) \right\} + \sum_{i=1}^{P} \log_2(1 - k_i^2) \]

\[ + \sum_{i=1}^{P} \gamma_i. \]

We can see from Eq. (10) that the total code length is a function of the prediction order P. In summary, the compression ratio of lossless coding depends on the autocorrelation (i.e., PARCOR coefficients) of input data. Therefore, we can neglect the first two terms and only consider variable terms of Eq. (10) as follows

\[ \tilde{C}(P) = \frac{N}{2} \sum_{i=1}^{P} \log_2(1 - k_i^2) + \sum_{i=1}^{P} \gamma_i \]

\[ = \sum_{i=1}^{P} \left( \frac{N}{2} \log_2(1 - k_i^2) + \gamma_i \right) \]

\[ = \sum_{i=1}^{P} c(i), \]

where \( c(i) \) (i = 1, 2, ..., P) denotes an estimated bit reduction for each prediction order. There is a trade-off between the bit-reduction term \( (N/2) \log_2(1 - k_i^2) \) and PARCOR bit-representation term \( \gamma_i \); the magnitude of PARCOR parameter \( k_i \) is always less than 1.0, meaning that the first term in Eq. (11) is negative (hence bit-reduction), while the PARCOR representation bit is positive. When \( N \) is large, \( C_e(P) \) becomes relatively small compared with \( C_e(P) \) and is negligible. On the other hand, when \( N \) is small, \( C_e(P) \) cannot be ignored. In such a case, the precision for PARCOR coefficients and prediction order \( P \) should be selected carefully. Thus, we precisely compare the influence of quantization error of PARCOR coefficients.

### 3.2. Non-Linear Bit Shift (NLBS) Quantizer

As described previously, there is a trade-off between
the bit-reduction term \(N/2\log_2(1 - k_i^2)\) and PARCOR bit-representation term \(\gamma_i\), depending on the quantizing precision. The quantizing error of PARCOR coefficients affects the compression performance. Here, the expected bit reduction \(\lambda_i\) in each sample is
\[
\lambda_i = \log_2(1 - k_i^2) \leq 0
\]  
where the constant value is avoided. Consequently, the inverse function can be assumed as
\[
\|k_i\| = \sqrt{1 - 2^\lambda_i} \approx 1 - 2^\lambda_i - 1 = \tilde{k}_i,
\]
and its partial differential becomes
\[
\frac{\partial \tilde{k}_i}{\partial \lambda_i} = -(\ln 2) \cdot 2^{\lambda_i - 1}.
\]

This means that conversely, if the magnitude of \(k_i\) is close to 0.0, the change in \(\lambda_i\) does not have significance. The interval of quantization follows power-of-two domain when one bit is added to the PARCOR representation term. A quantization algorithm based on this criterion can be constructed by bit shift operation as well as by inverted A-law/µ-law compression.

In fact, the PARCOR coefficients are calculated in 16-bit precision fixed-point arithmetic format (Q15) in G.711.0, and the maximum absolute value of integer PARCOR coefficients is limited to 32,750, i.e., \(\|k_i\| \leq (32,750/32,768) = 0.99945\). Where, Q15 format is a typical fixed-point representation that has one sign bit and 15 fractional bits to present the real numbers between \(-1\) and \(+1\) mapped into the integer values between \(-32,768\) and \(+32,767\). Here we defines \(K_i\) as an integer notation. The proposed non-linear bit shift (NLBS) quantization algorithm performs only the following several ITU-T basic operations [11–14].

1. The absolute integer values of PARCOR coefficients (Q15 format ‘sABC DEFG HIJK LMNO,’ where ‘s’ is dedicated to polarity sign, and each letter corresponds to zero or one) are defined as \(K_i^+ = \|K_i\|\). In other words, ‘sABC DEFG HIJK LMNO’ becomes ‘0ABC DEFG HIJK LMNO.’
2. Integer values of \(K_i^+\) are used for quantization and are arithmetically bit shifted to the right by \((16 - U_i)\), where \(U_i\) denotes the precision bits (let it be three here) for quantizing and depends on \(i\),
\[
W_i = [2^{-(16-U_i)}K_i^+]_i
\]
(i.e., ‘0ABC DEFG HIJK LMNO’ \(\rightarrow\) ‘0000 0000 0000 00AB’).
3. The absolute quantized PARCOR coefficient \(\|\tilde{k}_i\|\) is obtained by the bit shift operations \((16 - U_i - V_i - W_i)\)-right shifting and \((16 - U_i - V_i - W_i)\)-left shifting, where \(V_i\) is determined in advance (here, it is one):

\[
\|\tilde{k}_i\| = \begin{cases} 
0 & \text{if AB = 00, i.e., } W = 0, \\
1 & \text{if AB = 01, i.e., } W = 1, \\
2 & \text{if AB = 10, i.e., } W = 2, \\
3 & \text{if AB = 11, i.e., } W = 3,
\end{cases}
\]

(15) Finally, an appropriate value \(\epsilon\) is added for bias and a quantized PARCOR coefficient with polarity sign is provided as
\[
\hat{k}_i = \text{sgn}(K_i) \cdot \|\tilde{k}_i\| + \epsilon.
\]

In summary, sparse quantization is applied to values around zero, whereas absolute values close to one are quantized densely. To reduce complexity, the table look-up method is adopted in the G.711.0 implementation, thus this method needs only 6 weighted fixed-point operations in ITU-T basic operators.

3.3. Comparison of Bit Reduction

As described previously, quantizing PARCOR coefficients affects compression performance. Theoretically, the energy reduced by the \(P\)-th coefficient is
\[
E(P) = E(P - 1) \cdot (1 - k_p^2).
\]

When the quantized error of the coefficient is \(\Delta_p = k_p - \hat{k}_p\) (where \(\hat{k}_p\) is the quantized PARCOR coefficient), the energy changes to
\[
\hat{E}(P) = E(P - 1) \cdot (1 - k_p^2 + \Delta_p^2).
\]

According to Eq. (4), the energy of predicted residual signal obtained by the first quantized PARCOR coefficient is
\[
\hat{E}(1) = \sum_{n=1}^{N} (e_1(n))^2 \\
= \sum_{n=1}^{N} \{x(n) + (k_1 + \Delta_1) \cdot x(n - 1)\}^2 \\
\approx R(0) + 2(k_1 + \Delta) \cdot R(1) \\
+ (k_1^2 + 2k_1 \cdot \Delta_1 + \Delta_1^2) \cdot R(0) \\
= E(0) \cdot (1 - k_1^2 + \Delta_1^2),
\]

(19) where \(R(0) = \sum_{n=1}^{N} x(n)^2 \approx \sum_{n=1}^{N} x(n - 1)^2\), \(R(1) = \sum_{n=2}^{N} x(n) \cdot x(n - 1)\), and \(k_1 = -R(1)/R(0)\), as assumed in the Levinson-Durbin algorithm. Consequently, we must transcribe Eq. (11) as
\[
\hat{c}(1) = -\frac{N}{2} \log_2(1 - \hat{k}_1^2) + \gamma_1
\]
\[ \hat{c}(i) = \frac{N}{2} \log_2(1 - k_i^2 + \Delta_i^2) + \gamma_i. \] (20)

Here, \( \hat{c}(i) \) bears hat, because that it is the actual bit reduction estimated using the quantized PARCOR coefficients. Generally, when the quantization errors up to the \((i-1)\)-th order are ignored, we can assume the estimated bit reduction as described below:

\[ \hat{c}(i) = \frac{N}{2} \log_2(1 - k_i^2 + \Delta_i^2) + \gamma_i. \] (21)

We compared the proposed NLBS method and well-known conventional quantization methods (bit-shift, arcsin, and arctanh) \[8\], which are described below and the quantization errors are shown in Fig. 3.

- **Bit-shift quantizer (BS)**
  Integer PARCOR coefficient \( K_i \) is quantized Q-bit precision. Then, we can obtain \( \hat{K}_i \).

- **Arcsin quantizer**
  1. \( k_i = \arcsin(K_i/32,768) \).
  2. \( k_i \) is quantized to \( \hat{k}_i \) (Q-bit precision).
  3. \( \hat{K}_i = 32,768 \times \sin(\hat{k}_i) \).

- **Arctanh quantizer (Log-Area Ratio)**
  1. \( k_i = \tanh(K_i/32,768) \).
  2. \( k_i \) is quantized to \( \hat{k}_i \) (Q-bit precision).
  3. \( \hat{K}_i = 32,768 \times \tanh(\hat{k}_i) \).

- **NLBS quantizer (developed method)**
  Integer PARCOR coefficient \( K_i \) is quantized Q-bit precision (depends on \( U_i \) and \( V_i \)) with devised NLBS algorithm presented in the previous subsection. Then, we can obtain \( \hat{K}_i \).

Since the first and second PARCOR coefficients have unique distributions as shown in Fig. 4 and they are sensitive to the quantization in general, we show the results of \( \hat{c}(1) \) (Fig. 5) and \( \hat{c}(2) \) (Fig. 6), respectively. Here, both \( \gamma_1 \) and \( \gamma_2 \) correspond to the X-axis. We set the frame size 5 ms in 8 kHz sampling (i.e., \( N = 40 \)) because it is shortest frame length in G.711.0 therefore it becomes most severe and challenging condition in this experiment as described in Sub-sect. 3.1. We can estimate the result of longer frame case from Eq. (21), and the weight of bit-representation term \( \gamma_i \) decreases to half, quarter, etc. when the frame length is set to 80, 160 and so on. Input speech signal was
Table 1 Comparison of estimated bit reduction for approximated arcsin quantizers that need still floating-point operation and devised NLBS with fixed-point implementation.

<table>
<thead>
<tr>
<th>Quantization method</th>
<th>Complexity (# of operations)</th>
<th>Minimum value of average of ( \hat{c}(1) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>arcsin (w/o approximation)</td>
<td>N/A</td>
<td>-42.46</td>
</tr>
<tr>
<td>arcsin (4th polynomial Taylor series)</td>
<td>7@floating-point</td>
<td>-42.22</td>
</tr>
<tr>
<td>( x + (1/6) \cdot x^3 + (3/40) \cdot x^5 + (5/112) \cdot x^7 )</td>
<td>43@fixed-point</td>
<td>-42.22</td>
</tr>
<tr>
<td>arcsin (3rd polynomial Taylor series)</td>
<td>6@floating-point</td>
<td>-42.00</td>
</tr>
<tr>
<td>( x + (1/6) \cdot x^3 + (3/40) \cdot x^5 )</td>
<td>26@fixed-point</td>
<td>-42.00</td>
</tr>
<tr>
<td>arcsin (2nd polynomial Taylor series)</td>
<td>3@floating-point</td>
<td>-41.95</td>
</tr>
<tr>
<td>( x + (1/6) \cdot x^3 )</td>
<td>17@fixed-point</td>
<td>-42.23</td>
</tr>
<tr>
<td>NLBS (developed method)</td>
<td>6@fixed-point</td>
<td>-42.23</td>
</tr>
</tbody>
</table>

Table 2 Comparison of file size in octets and compression ratio in percent (Compressed-size/input-size).

<table>
<thead>
<tr>
<th>Input</th>
<th>6,113,280 (100%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Full search</td>
<td>2,807,954 (45.9%)</td>
</tr>
<tr>
<td>Estimation</td>
<td>2,837,714 (46.4%)</td>
</tr>
</tbody>
</table>

\[
\log_2(1 - k_i^2) \approx -\alpha \cdot k_i^2. \tag{22}
\]

As a result, the optimum prediction order \( P_{opt} \) can be obtained by finding the approximated minimum total code length

\[
P_{opt} = \arg \min_P C(P)
\]

\[
= \arg \min_P \sum_{i=1}^{P} \left( \frac{N}{2} \log_2(1 - k_i^2) + \gamma_i^{(P)} \right)
\]

\[
\approx \arg \min_P \sum_{i=1}^{P} \left( -\alpha \cdot \frac{N}{2} k_i^2 + \gamma_i^{(P)} \right), \tag{23}
\]

where \( C(P) \) comes from Eq. (11). Using this method, costly logarithmic operation is avoided. While a floating-point operation, the logarithmic function needs 25 weights of ITU-T basic operators. Contrary, the modified version uses only 2 weights in a fixed-point architecture. Although the value of constant theoretically should be 1.44, we determined that it is 2.1 for the \( N = 40 \) sample case in a pilot study.

4. LOW-COMPLEXITY PREDICTION-ORDER ESTIMATOR

4.1. Estimation of an Optimum Prediction Order

As shown in the previous section, there is also a trade-off between bit-reduction term \((N/2) \log_2(1 - k_i^2)\) and PARCOR bit-representation term \( \gamma_i \), depending on the number of prediction orders. For the 40-sample case, the maximum prediction order is set to four in G.711.0. When the prediction order is low, not much bit reduction is expected. When the order is high, the first PARCOR coefficient can be expected to be close to one. Then, the PARCOR bit-representation term becomes \( \gamma_i^{(P)} \) with suffix \( P \) because the bits for quantization are different among and dependent on the prediction orders.

The actual prediction order is searched for the minimum estimated code length. The logarithmic function of the bit-reduction term is approximated by the Maclaurin series as obtained from ITU-T P.501 and these signal was resampled to narrow-band data.

As seen from these graphs, the proposed method achieves bit reduction similar to the arcsin quantizer. Actually, the conversion of arcsin/sin and arctanh/tanh was performed in floating-point precision in this experiment, so the results of these two methods can be regarded as the “best case.” The arcsin quantizer achieves maximum bit reduction in the case of the first prediction order. However, it was degraded when the function was approximated using Taylor-series for complexity reduction, as shown in Table 1. To keep the accuracy of variables, one floating-point operation needs several fixed-point operations. For example, one floating-point multiplication requires at least four fixed-point operations (two multiplications for sign and mantissa, an addition for exponent, and an extract). Finally, the developed NLBS quantizer yields better performance in terms of the trade-off between bit reduction and complexity.

4.2. Experimental Evaluation

Table 2 shows the difference of total output file size between the minimum octet (8 bits) and the estimated octets with the prediction order \( P_{opt} \) for the 40-sample case. Here, the minimum means that the hacked G.711.0 encoder processes all stages, including prediction filtering, entropy coding and so on, for all four order cases, then the actual minimum code length is selected (i.e., brute force full search). The same speech signal described in the previous
section was used. The difference of compression ratio between full-search and the devised estimation method is only 0.5%. Moreover, the maximum increase of the bits caused by mischoice (i.e., estimation error) is within two octets for most of frames (more than 99%) as shown in Fig. 7. The developed estimator finds the optimum prediction order for most frames (about 85%) though it needs only a few operations.

5. CONCLUSION

A tool that efficiently quantizes the PARCOR coefficients and based on estimation of the optimal prediction order is proposed. The quantization of PARCOR coefficients is based on a criterion that minimizes the entropy of the prediction residual signal and the prediction order is determined by the values of PARCOR coefficients by using the minimum description length criterion. To reduce the complexity of PARCOR quantizer, a novel approximation scheme was also developed. By combining these methods, a better tradeoff between compression performance and complexity was achieved. The tool is used in the new ITU-T recommendation G.711.0, the lossless compression technology of the G.711 payload. It is expected that G.711.0 will widely be used for the efficient transmission of the most deployed speech coding benchmark, G.711.

REFERENCES

Yutaka Kamamoto received the B.S. degree in applied physics and physico-informatics from Keio University in 2003, and the M.S. and Ph.D. degrees in information physics and computing from the University of Tokyo in 2005 and 2012, respectively. Since he joined NTT Communication Science Laboratories in 2005, he has been studied signal processing and information theory, especially lossless coding of time-domain signal. He received the Awaya Prize Young Researcher Award from the Acoustical Society of Japan (ASJ) in 2011. He is a member of ASJ, the Institute of Electronics, Information and Communication Engineers of Japan (IEICE), the Information Processing Society of Japan (IPSJ), and the Institute of Electrical and Electronics Engineers (IEEE).

Takehiro Moriya received the B.S., M.S., and Ph.D. degrees in mathematical engineering and instrumentation physics from the University of Tokyo, in 1978, 1980, and 1989, respectively. Since joining NTT laboratories in 1980, he has been engaged in research on medium to low bitrate speech and audio coding. In 1989, he worked at AT&T Bell Laboratories, Murray Hill, NJ, as a Visiting Researcher. Since 1990 he has contributed to the standardization of coding schemes for the Japanese Public Digital Cellular system, ITU-T, and ISO/IEC MPEG. From 1997 to 2001, he was a member of the IEEE Speech Technical Committee. He is a member of ASJ, IEICE, IPSJ, IEEE, and the Audio Engineering Society (AES).

Noboru Harada received the B.S. and M.S. degrees from the Department of Computer Science and Systems Engineering of Kyushu Institute of Technology, in 1995 and 1997, respectively. He joined NTT Laboratories in 1997. His main research area has been lossless audio coding, high-efficiency coding of speech and audio, and their application. He is an editor of ISO/IEC 23000-6:2009 Professional Archival Application Format, ISO/IEC 14496-5:2001/Amd.10:2007 reference software MPEG-4 Audio Lossless Coding, and ITU-T Recommendation G.711.0 Lossless compression of G.711 pulse code modulation. He is a member of ASJ, IEICE, IEEE, and AES.

Yusuke Hiwasaki obtained B.E. in instrumentation engineering, M.E., and Ph.D. degrees in computer science, from Keio University, in 1993, 1995, and 2006, respectively. Since joining NTT Human Interface Laboratories (now Media Intelligence Laboratories) in 1995, he has been engaged in research field of low bit-rate speech coding and voice-over-IP telephony. From 2001 to 2002, he was a guest researcher at Kungliga Tekniska Hogskolan (Royal Institute of Technology) in Sweden. Since 2007, he has been active in standardization of speech coding especially at ITU-T Study Group 16 (SG16) and acted as the editor of Recommendation ITU-T G.711.1. From 2009, he had been Associate Rapporteur of ITU-T SG16 Q.10, a question on speech coding matters, and then became Rapporteur in 2011. He is also a member of ASJ, IEICE, and IEEE.