Coding Method for Successive Pitch Periods

Ari Heikkinen, Vesa T. Ruoppila, and Samuli Pietilä

Nokia Research Center, Speech and Audio Systems Laboratory
P.O. Box 100, FIN-33721 Tampere, Finland

Abstract

This paper presents a coding method for successive pitch periods of a speech signal. In the proposed method, a priori knowledge of the statistical properties of successive pitch periods is used by designing a shaped lattice structure that covers the most probable points in the pitch space. We also present briefly a pitch search algorithm for the shaped lattice. An example implementation of shaped lattice coding is given for a modified IS-641 speech coder. Based on the simulation results, the proposed scheme achieves capacity savings compared to the conventional methods.

1. Introduction

Based on the human speech processing mechanism the pitch period contour of voiced speech evolves slowly in time [1]. To exploit this phenomenon several approaches have been proposed to increase the coding efficiency. This is especially true for the speech coders transmitting the pitch information on a subframe basis, such as code excited linear prediction (CELP) coders.

In [2] restrictive pitch deviation coding is proposed where an average pitch period for a frame is first computed and coded, and then for each subframe the optimal pitch period is found around the average value. The optimal pitch values are represented using differential coding with respect to the average pitch period. In many current CELP coders such as the IS-641 coder [3], the absolute pitch period is transmitted for certain subframes while the remaining dimensions represent differences between successive pitch periods, referred to as a delta period, is transmitted for the other subframes. Generally, the delta periods may attain uniformly distributed values from a limited range facilitating their coding. This can be interpreted as a multi-dimensional rectangular lattice populated uniformly by points that define the delta periods over the frame.

In [4], the statistical properties of the delta periods are exploited by using Huffman coding, and the saved bits are used to enhance the innovation codebook. In [5], a subset consisting of the most probable pitch ranges is determined for each subframe depending on the previous transmitted pitch value. The subsegments are experimentally defined typically consisting of the neighborhood of the previous pitch value together with its multiples and submultiples for voiced speech and equally spaced pitch values for unvoiced speech.

In this paper, the emphasis is put on efficient coding of successive pitch periods during voiced speech. In contrast with approaches that employ a multidimensional rectangular lattice for the delta periods, we utilize a priori knowledge on the distribution of successive pitch values for designing a shaped lattice. The shaped lattice covers only the region where the successive delta periods are most likely located. In this paper, we examine only a shaped lattice that is a union of hypercubes which all may have a different resolution for pitch periods. This simplistic lattice structure allows simple design and indexing for representing the lattice point, and thus facilitates the encoding and decoding.

Shaped lattices render the conventional search methods of pitch period impractical as usually the search progresses subframe wise in closed-loop. Therefore a new search method for the shaped lattice is needed. We will introduce this in Section 2 with the general principles of the proposed coding method. An example implementation for a modified IS-641 coder and the objective test results are discussed in Section 3. In Section 4 conclusions are drawn.

2. Shaped Lattice

In typical CELP coders the coding of successive pitch periods can generally be described as an n-dimensional lattice where each dimension represents a pitch period in a corresponding subframe. Usually, the first dimension of a lattice represents the absolute pitch period in the first subframe while the remaining dimensions represent differences between successive pitch periods over the frame.

In most speech coders that utilize differential coding, the lattice structure for n delta periods is described as a set of points with a regular arrangement in an n-dimensional space. Besides of their regularity, another key feature of the current lattices is the rectangular shape of their projection onto a two-dimensional plane. The structure of the lattice is usually constant regardless of the pitch period in the previous segment.

Figure 1 illustrates coding of delta periods $d_1$ and $d_2$. The value ranges of $d_1$ and $d_2$, $d_{1\min} \leq d_1 \leq d_{1\max}$, $d_{2\min} \leq d_2 \leq d_{2\max}$, result a two-dimensional rectangular lattice $L$. The lattice covers all possible combinations of $d_1$ and $d_2$ between their minimum and maximum values. The minimum and maximum values of the $i$th delta period $d_i$ are denoted by $d_{i\min}$ and $d_{i\max}$, respectively. While the lattice in Figure 1 is two-dimensional, higher dimensional lattices can be derived by a straightforward extension of the two-dimensional case.

Once the shape and the region of the lattice are defined, an important parameter is the density of the lattice. Naturally, the bit rate is a monotonically increasing function of the density. The density of the lattice quantizer is defined by the
accuracy used for pitch period information. Normally, fractional values are used to improve the quality of the synthesized speech.

In a typical lattice coding of delta periods, attention is only paid to the selection of \( d_{\text{min}} \) and \( d_{\text{max}} \), while the rectangular shape of the lattice is maintained. No further care is taken to describe how a suitable set of points is chosen to cover only the most likely points used. Since the pitch period evolves usually smoothly during voiced speech, the rectangular lattice covers also points that are used rarely. Thus, the coding efficiency can be increased by shaping the lattice to eliminate unlikely pitch period combinations from the resulting coding scheme. Furthermore, regions with different point density can be defined within the shaped lattice for representing a varying fractional resolution of pitch period. This can be useful at pitch onsets, for example, where both the accurate pitch information and the possibility for sudden jumps of pitch period are important features to achieve a good performance.

The general principle of a shaped lattice \( S \) is presented in Figure 1, where the union of two sublattices with different point densities, denoted by \( S_1 \) and \( S_2 \), defines the shaped lattice structure. Points close to \( (d_{1\text{min}}, d_{2\text{min}}), (d_{1\text{min}}, d_{2\text{max}}), (d_{1\text{max}}, d_{2\text{min}}), \) and \( (d_{1\text{max}}, d_{2\text{max}}) \) have been left out from the shaped lattice \( S \) as they occur rarely in voiced speech. Thus they can be discarded from the shaped lattice with minor effect on the resulting speech quality. On the other hand, higher point density in the sublattice \( S_1 \) allows to utilize a finer pitch resolution when pitch period evolves smoothly without increasing significantly the bit rate. In this figure, the shaped lattice is a subset of lattice \( L \) but generally this is not necessary.

Because of the closed-loop structure of most coders utilizing differential coding of the pitch period, the optimal index search in a lattice is done subframe wise. Thus the search proceeds sequentially along one coordinate axis of the lattice in time. Generally this is done in the following way. Firstly, an open-loop pitch period estimate is determined for the subframes containing the absolute pitch period and the following delta periods. Typically, integer values are used in open-loop search for reduced complexity. Thereafter, the optimal index search is done in closed-loop fashion sequentially for each dimension. For the first subframe this is done in the neighborhood of the selected open-loop pitch period while the search area for the other subframes consists of the neighborhood of the previously selected pitch period.

With shaped lattices this approach is not necessarily optimal since the possible set of lattice points in each dimension depends heavily on the selected point in the previous dimension. Therefore, an open-loop estimate point in the shaped lattice is first determined in the multi-dimensional space. After that, the optimal index in each dimension is found in a closed-loop fashion in the neighborhood of the estimated open-loop point one dimension at a time. This method is illustrated in Figure 1, where the estimated point is denoted by a black dot. The closed-loop search examines the points that belong to the intersection of the shaped lattice \( S \) and the search region \( C \) centered to the open-loop pitch estimate. The index determined by the closed-loop search defines uniquely the pitch period over the subframes covered by the lattice.

The lattice shape affects the trade-off between coding efficiency and the implementation issues of pitch search and indexing algorithms. Conventional rectangular lattices allow simple search and indexing compared to shaped lattices as the subframes can be considered independently of each other. The implementation advantages of rectangular lattices can be combined with the coding efficiency of shaped lattices when the shaped lattice is composed as a union of rectangular sublattices. This approach will be considered in more detail in the next section.

3. Example Implementation for IS-641

As an example, we defined and implemented a shaped lattice coding for the IS-641 speech coder. In the IS-641 speech coder the long term prediction (LTP) analysis and coding is done in a quite similar way as in many current low bit rate CELP coders. A 20-ms speech frame is divided into four subframes of equal length. Pitch period is searched with a modified autocorrelation method for every subframe. For the first and third subframe the pitch period is searched from the range of \( 19^1/3 \) to \( 143 \) samples. In the range of \( 19^1/3 \) to \( 84^1/3 \) samples, a resolution of \( 1/3 \) is used while integer values are used in the range of \( 85 \) to \( 143 \) samples. For the second and fourth subframes the pitch periods are searched from the neighborhood of the pitch periods in the previous subframes. The range of the search is \( -4^1/3 \) to \( 5^1/3 \) samples using a resolution of \( 1/3 \). The pitch period in the first and third subframe is coded with 8 bits while the delta period in the second and fourth subframes is coded with 5 bits, resulting in 25 bits per frame. In the example implementation we modified the original method such that the absolute pitch period is used only for the first subframe while delta pitch periods are used for the other subframes.

Our test material consisted of 39434 frames of American-English speech spoken by multiple talkers. Figure 2 represents distribution of the delta periods derived from the voiced speech segments using the modified IS-641 speech coder. The delta period range is \( \pm 6 \) samples. The difference between the pitch periods of the \((i+1)\)th subframe and the \(i\)th subframe is denoted by \( d_i \). The delta periods are rounded to integer values in the figure even though \( 1/3 \) resolution was used in this simulation.
As we can see from Figure 2, a shaped lattice structure is justified, since the combinations of two large delta values are rare. Based on the empirical data presented in this figure, a four-dimensional lattice structure was designed representing the four subframes of the modified IS-641 speech coder. The first dimension corresponds to the pitch period in the first subframe while the other dimensions represent the differences between successive pitch periods. A lattice structure shown in Figure 3 was derived for the three delta periods, denoted by $d_1$, $d_2$, and $d_3$. In our example, the three-dimensional lattice for delta periods was kept constant regardless of the pitch period in the first subframe. However, the definition of the lattice described below allows a more general structure if needed. In the figure the regularly distributed points in the lattice are denoted by circles.

In Figure 3, the lattice is composed of a union of non-overlapping hypercubes $D_i$, which are defined by the delta period range and the resolution used in each dimension. Different hypercubes are marked by dashed lines in the figure, and can be defined by their unique edges. For example, the hypercube $D_2$ is defined by the edges $d_2$, $d_2$, and $c_3$. In our implementation, the structure of the lattice was described by the hypercube matrix $D$. Each row of $D$ defined a unique four-dimensional hypercube such that the $i$th row was

$$D(i,:) = [p_{i,\text{min}} \ p_{i,\text{max}} \ r_{i,0} \ d_{i,1,\text{min}} \ d_{i,1,\text{max}} \ r_{i,1} \ \cdots \ d_{i,2,\text{min}} \ d_{i,2,\text{max}} \ r_{i,2} \ d_{i,3,\text{min}} \ d_{i,3,\text{max}} \ r_{i,3}]$$

where $p_{i,\text{min}}$, $p_{i,\text{max}}$, and $r_{i,0}$ define the pitch period range and the resolution for the first subframe. The ranges of delta periods in the last three subframes are defined by $d_{i,\text{min}}$ and $d_{i,\text{max}}$, where $j$ is the subframe index. The corresponding resolution in each subframe is denoted by $r_j$.

With the lattice structure described above, the encoding process is quite straightforward. For encoding the index of a certain point in the lattice we define the starting index, and the number of points in each unique edge of every hypercube. The encoding process starts by determining the index of the hypercube into which the found pitch period combination $(p, d_1, d_2, d_3)$ belongs. The hypercube $D_i$ containing this point fulfills all the following inequalities:

$$p_{i,\text{min}} \leq p \leq p_{i,\text{max}},$$
$$d_{i,j,\text{min}} \leq d_j \leq d_{i,j,\text{max}},$$

for $j = 1, 2, 3$.

Next, the index of the point in the corresponding hypercube is defined. This is done by first defining the coordinates of each dimension inside the hypercube $D_i$. The coordinate $p_j$ for the $(j+1)$th subframe is given by

$$p_j = (p - p_{i,\text{min}}) r_{i,0},$$
$$p_j = (d_j - d_{i,j,\text{min}}) r_{i,j},$$
for $j = 1, 2, 3$.

Thus, the index $s$ of the point $(p, d_1, d_2, d_3)$ in the lattice can be determined as

$$s = s_{D_0} + p_0 + p_1 r_{i,0} + p_2 r_{i,1} r_{i,0} + p_3 r_{i,2} r_{i,1} r_{i,0},$$

where $s_{D_0}$ is the offset of the hypercube $D_0$. The number of points in each edge of $D_i$ in the $(j+1)$th dimension is denoted by $n_{i,j}$.

After describing the lattice in a suitable way, the next issue is to find the appropriate boundary values for it. In our example, we coded the first dimension in a similar way as in the IS-641 coder while the shaped lattice of Figure 3 was used for the other dimensions. Furthermore, the lattice structure used was symmetric with respect to axis $d_1$, $d_2$, and $d_3$. The point distribution in the last three dimensions was uniform and 1/3 resolution was used. Because of the symmetry, the three-dimensional lattice can be unambiguously defined by one corner point of the projection of $D_0$ to axis $d_1$ and $d_2$, see Figure 3. The optimal index was searched in the shaped lattice by first estimating the index jointly in each dimension in an open-loop fashion using integer resolution. This open-loop estimate was refined using closed-loop search sequentially in each dimension.

In our simulations, three different shaped lattices $S_A$, $S_B$, and $S_C$ were implemented with corner points $(2/3, 1/3, 2/3, 2/3)$, $(2/3, 2/3, 2/3, 2/3)$, and $(1/3, 2/3, 2/3, 2/3)$, respectively. As a reference, two cubic lattices $L_A$ and $L_B$ with maximum delta periods of $2/3$ and $2/3$ were used. These ranges were selected based on the distributions presented in Figure 2. The simulation results are presented in Table 1. The results are expressed as segmental signal to noise ratios (SegSNR) between the voiced sections of the input speech and synthesized speech, together with the

![Figure 2. The differences between successive pitch periods in the modified IS-641 speech coder.](image-url)
number of bits needed for the coding of the delta periods in each frame. A segment length of 64 samples was used and silent segments were discarded in the SegSNR computation. The speech sample used in all simulations consisted of sentences spoken by four male and four female talkers in clean conditions. The total length of the sample was 10000 frames. As it can be seen from the table of results, the coding efficiency of successive pitch periods can be increased by using the shaped lattice structure.

Table 1: Segmental SNR and the number of bits needed for different pitch coding schemes.

<table>
<thead>
<tr>
<th></th>
<th>segSNR (dB)</th>
<th>bits</th>
</tr>
</thead>
<tbody>
<tr>
<td>L A</td>
<td>8.24</td>
<td>12.26</td>
</tr>
<tr>
<td>L B</td>
<td>8.09</td>
<td>10.38</td>
</tr>
<tr>
<td>S A</td>
<td>8.28</td>
<td>11.78</td>
</tr>
<tr>
<td>S B</td>
<td>8.11</td>
<td>10.00</td>
</tr>
<tr>
<td>S C</td>
<td>8.05</td>
<td>9.17</td>
</tr>
</tbody>
</table>

4. Conclusions

In this paper, a shaped lattice coding for successive pitch periods together with its search routine was presented. Based on the simulations using the modified IS-641 speech coder, it can be concluded that the proposed method improves the efficiency of pitch coding compared to the conventional method using rectangular lattice structures.

5. References