BENEFIT AND COST ANALYSIS OF USING THE IMPROVED VECTOR QUANTIZER DESIGN ALGORITHM FOR GLOTTAL SOURCE WAVEFORM COMPRESSION

Peter Veprek 1,2 and Alan B. Bradley 2

1 Panasonic Speech Technology Laboratory, Santa Barbara, CA, U.S.A.
2 Faculty of Engineering, RMIT University, Melbourne, Australia
pveprek@research.panasonic.com,alanb@rmit.edu.au

ABSTRACT
Vector quantizer design is an essential step required in the development of many quantization (compression) and clustering (classification) tasks. The iterative codevector readjustment and reassignment algorithm (ICRRA) is shown to outperform the traditional generalized Lloyd algorithm for the task of glottal source waveform compression. Analysis of the results reveals that the ICRRA offers improved performance at a very high confidence level. The improvement is achieved at the expense of increased computation. Storage requirements of the algorithm, however, remain virtually unchanged for typical conditions. Overall, the ICRRA is particularly suitable for off-line processing where longer execution time is well justified by the benefits of reduced reconstruction distortion.

1. INTRODUCTION
Recently, an improved vector quantizer design algorithm was proposed and its superiority to several established procedures was demonstrated for typical marginal distributions in the case of an artificial data source [1]. The purpose of this paper is twofold. The first objective is to provide an evaluation of the proposed iterative codevector readjustment and reassignment algorithm (ICRRA) for the task of glottal source waveform compression. The second objective is to determine the benefit provided by the ICRRA and the cost in terms of increased computation and storage requirements. Such analysis will provide a decision tool for researchers and practitioners seeking improved performance in various quantization and clustering tasks.

2. REVIEW OF ICRRA
Operation of the ICRRA can be summarized as follows. After generating the initial codebook, the algorithm first performs the standard generalized Lloyd algorithm (GLA), i.e. repetitive partitioning of the training data set using the nearest neighbor condition and subsequent codebook update by applying the centroid condition [2] [3]. Next, the algorithm identifies cells with the smallest and largest average distortion of the associated data vectors. The codevector of the cell with the smallest distortion is then freed by amalgamating the cell with its nearest neighbor. The created empty slot in the codebook is populated by splitting (small perturbation) the codevector of the cell with the largest distortion. The entire process is then repeated as illustrated in Fig. 1. The codevector reassignment has the effect of reducing the reconstruction distortion caused by “poorly-fitting” cells and as a result enhances the quality of the generated codebook thus improving the quantization performance.

3. GLOTTAL SOURCE WAVEFORM COMPRESSION
3.1. Speech corpus and the glottal source waveform
The effectiveness of the ICRRA was evaluated for the task of glottal source waveform compression. The evaluation speech corpus comprised a set of 6,708 units - diphones and demisyllables [4]. Each unit was parameterized using a method described in [5] into a series of pitch-synchronous pulses with source-filter representation. For unvoiced speech the pulse length was universally set to 5 ms. Source data included a time-domain glottal excitation waveform and filter data consisted of (five) formant frequencies and bandwidths. The glottal waveform
was obtained by inverse filtering using the estimated filter model. Filter parameters were iteratively optimized by minimizing a cost function (arc-length minimization) yielding continuous formant tracks and consistent pitch epoch marking [5]. The analysis resulted in a total of 228,820 pulse entities out of which 177,457 (77.6%) were voiced and 51,363 (22.4%) were unvoiced. In addition, pulses were also categorized according to the region in which they occurred. 67,879 (29.7%) pulses were characterized as optional - belonging to a stable region of a phone nucleus. The overall set of pulses spanned the full 25,828,475 quarters were left for the more diverse unvoiced pulses. This ratio remained essentially unchanged throughout the ICRRA. It was observed that the optional (stable region) pulses were usually slowly evolving and were good candidates for pulse repetition and pulse interpolation. Therefore, the critical pulses were given a preference when updating codevectors of the codebook during the iterative codebook refinement, data vectors nearest the centroids were taken. The motivation for this was the hypothesis that data vector (i.e. the glottal pulse) interpolation in the time domain would introduce additional distortion and consequently reduce naturalness of the reconstructed speech. An additional benefit gained was a substantial increase in the execution speed achieved by keeping track of the codevectors changed since the last iteration of the ICRRA.

Several heuristics were applied as further constraints of the quantization process. The objective here was both to improve and speed-up the quantization. A quarter of the initial codebook was allocated for voiced pulses while the remaining three quarters were left for the more diverse unvoiced pulses. This ratio remained essentially unchanged throughout the ICRRA. It was observed that the optional (stable region) pulses were usually slowly evolving and were good candidates for pulse repetition and pulse interpolation. Therefore, the critical pulses were given a preference when updating codevectors of the codebook by applying a penalty factor of 2 to the more uniform optional pulses during computation of vector distance. In addition, several conditions were tested prior to computation of the $SD_p$. Only pulses of the same voicing type and only pulses of similar pitch period in the case of voiced speech were ever compared. The same constraints were used for all evaluated codebook sizes reported in the next section.

4. Experimental Evaluation Results

4.1. Experiment 1 – $SD_p$ improvement and its significance

An experimental evaluation was carried out in which individual pulses of the glottal source waveform were quantized using the described constraints. The vector quantization process was repeated six times for each of several different codebook sizes ranging from 256 to 2,048 codevectors. Each time, a new randomly selected initial codebook was used. Although more repetitions would increase the statistical confidence of the results, prior investigation in [1] demonstrated that the ICRRA greatly reduces the effect of the initial codebook on the quantization performance and therefore relatively few repetitions are required in practice to obtain representative results. The earlier findings were also confirmed by the results of this evaluation, where the $SD_p$ variance of the ICRRA was greatly reduced compared to GLA. Fig. 4 depicts quantization performance measured by the $SD_p$ for varying codebook sizes. The performance is shown as the relative improvement of ICRRA over GLA compared to GLA over randomly selected codebook (Rand). The relative improvement was computed as the mean ($SD_p^{GLA} - SD_p^{ICRRA}$) / ($SD_p^{Rand} - SD_p^{GLA}$). On
average in this experiment, ICRRA gave 10% improvement over GLA.

<table>
<thead>
<tr>
<th>Codebook size</th>
<th>SDp improvement</th>
</tr>
</thead>
<tbody>
<tr>
<td>256</td>
<td>9.7%</td>
</tr>
<tr>
<td>512</td>
<td>8.6%</td>
</tr>
<tr>
<td>1024</td>
<td>9.8%</td>
</tr>
<tr>
<td>2048</td>
<td>12.1%</td>
</tr>
</tbody>
</table>

Figure 4. Comparison of quantization performance.

4.2. Experiment 2 – effect of additional iterations on SDp

In the above experiment, the number of times the codevector reassignment could be performed was limited to \(4+N/32\), where \(N\) is the codebook size (i.e. about 3% of the codebook size). For each reassignment, a maximum of 8 readjustments were allowed. This setting was based on (a) the requirement that the reassignment loop did not increase the execution time of the algorithm more than approximately ten times and (b) the observation that the readjustment loop usually converges to the local minimum of the SDp within about five iterations. In this experiment, the maximum number of reassignment and readjustment iterations was increased to \(4+N/2\) and 16, respectively. These maximums provided sufficient scope to confirm the trend of reducing incremental improvement as the reassignment and readjustment iteration counts increase.

In Fig. 5, the SDp (for codebook size 1,024) is plotted as a function of time. The time is shown as a factor relative to GLA (i.e. GLA = 1.0). The ICRRA reached a minimum SDp in the 1st readjustment iteration of the 130th reassignment iteration. A total of 266 readjustment iterations were performed, i.e. on average there were just over 2 readjustments per reassignment undertaken. Further iterations did not improve the quality of the codebook as both the codebook definition and SDp entered an oscillatory limit cycle beyond this point. The 130 reassignment iterations correspond to 12.7% of the codebook size, perhaps suggesting the number of effective readjustment iterations \(4+N/8\) for the described conditions. It is worthy noting, however, that the allowed maximum number of both the readjustment and reassignment iterations can be set arbitrarily high because several termination conditions including detection of the described oscillation can easily be applied to optimally terminate both types of iteration. It can be seen that, in this case, the ICRRA greatly benefited from the increased number of iterations and yielded a result significantly better than the previous experiment at the expense of increased execution time. In this case, the final value of SDp was 4% lower compared to the previous experiment for the same codebook size. Table 1 lists the SDp value and its improvement relative to Rand and GLA, respectively for experiment 2. The relative improvement shows that the GLA reduced the distortion by 13.0% compared to Rand, while the ICRRA achieved 19.3% reduction. In relative terms, the ICRRA reduced the final distortion achieved by the GLA by 7.2%. The ICRRA-over-GLA improvement is 48% of the GLA-over-Rand improvement, which is a considerable amount.

Table 1: Comparison of SDp improvement for GLA and ICRRA.

<table>
<thead>
<tr>
<th>Method</th>
<th>SDp [dB]</th>
<th>Relative to Rand</th>
<th>Relative to GLA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rand</td>
<td>0.190869</td>
<td>0.0%</td>
<td>–</td>
</tr>
<tr>
<td>GLA</td>
<td>0.165993</td>
<td>13.0%</td>
<td>0.0%</td>
</tr>
<tr>
<td>ICRRA</td>
<td>0.154039</td>
<td>19.3%</td>
<td>7.2%</td>
</tr>
</tbody>
</table>

5. Benefit and Cost Analysis

5.1. Benefit of reduced quantization distortion

Inspection of the evaluation results seen in Fig. 4 clearly shows the advantage of the ICRRA compared to the GLA. In all cases, the ICRRA outperformed the GLA. In order to determine the statistical significance of the improvement, several hypotheses concerning the improvement value at different confidence levels were tested and quantified using a paired t-test [8] in the case of codebook size equal to 1,024. An additional null hypothesis was tested that determined confidence for the statement that the ICRRA yields an improvement. The result of this analysis shows that the ICRRA outperforms the GLA with >99.9% certainty for the described conditions (i.e. \(4+N/32\) and 8 iterations). There is >99% certainty of achieving 52.3% of the mean improvement. For more iterations, even better results can be obtained as illustrated in Fig. 5, where the SDp improvement value reached 0.011954 dB after 130 reassignment iterations. The improvement of the ICRRA over the GLA achieved in experiment 2 through additional iterations of the algorithm more than doubled that achieved in experiment 1.

5.2. Cost of increased computation and storage

The ICRRA requires more computation compared to the GLA. The codebook quality can be traded off against computation time depending on the application. Fig. 5 depicts how the
reconstruction distortion can be reduced at the expense of additional iteration of the algorithm. The increase in computation time is significant and may be prohibitive in the case of particular real-time applications. However, many off-line quantization tasks, such as data storage, can take full advantage of the ICRRA to optimize their performance. Table 2 shows relative break-up of the processing time into 5 major operations for the case depicted in Fig. 5. The most demanding task is the nearest neighbor search taking up more than half of the computation time. The second most time consuming task is the codevector reallocation using over a third of the time. The remaining three operations account for in excess of a tenth of the time. The required computation time could be reduced, for instance, by applying additional constraints of the quantization process, by the use of hierarchical quantization, and/or by any of the known optimization techniques. It is worthy noting that the cost of increased computation time is to a certain degree compensated by the fact that the ICRRA is less sensitive to the initial codebook than the GLA. Consequently, the ICRRA is particularly suitable to off-line quantization tasks where the ICRRA may typically be performed several times, each time with a different initial codebook.

Table 2: Allocation of time spent on different tasks.

<table>
<thead>
<tr>
<th>Task</th>
<th>Time allocation</th>
</tr>
</thead>
<tbody>
<tr>
<td>nearest neighbor search</td>
<td>54.7%</td>
</tr>
<tr>
<td>codevector readjustment</td>
<td>33.5%</td>
</tr>
<tr>
<td>cell centroid computation</td>
<td>9.9%</td>
</tr>
<tr>
<td>unused codevector filling</td>
<td>4.2%</td>
</tr>
<tr>
<td>codevector reallocation</td>
<td>1.6%</td>
</tr>
</tbody>
</table>

Compared to the GLA, the ICRRA also requires additional storage space to store the number of data vectors in each cell and the total distortion within the cell in order to perform the codevector reallocation. The storage space is proportional to the codebook size and represents only a small increase in the memory requirements of the algorithm, i.e. one extra integer and one floating point number per cell. This cost could be reduced by representing the cell distortion using an integer number and by compact bit packing.

Because, unlike GLA, the ICRRA is not guaranteed to iterate monotonically to a better solution during the codebook refinement, it needs to keep track of the best codebook encountered so far. Given the described constraints, the best tentative codebook can be represented by simply storing indexes of the data vectors representing every cell. Again, this cost is proportional to the codebook size and log-proportional to the amount of training data. For the evaluation corpus used and codebook size of 1,024 this cost represents only about 3 kB (1,024 x 24 bits).

6. DISCUSSION

In the current implementation of the ICRRA, only one codevector is reassigned at a time. It might be possible to reduce the number of computations required by the algorithm by reassigning several codevectors at the same time. However, such an operation would impact computation of the distortion reduction estimate (11) particularly in the case when the candidate cells are adjacent. A problem also arises in the case when several adjacent cells need to be merged or split.

Performance of the source waveform quantization was also evaluated subjectively by a series of listening tests. In these tests, speech resynthesized from the compressed data was compared to a baseline system in which all source pulses were simply stored (i.e. perfectly reconstructed). Additionally, resynthesized speech produced for different compression ratios was also compared. The subjective tests concluded that the method provides very high quality of the reconstructed speech for codebook sizes in the order of a few thousand to ten thousand pulses. Interestingly, the quality of voiced speech was judged as nearly transparent using only a few hundred voiced pulses in the pulse codebook. Therefore, future work will focus on improvement of the technique for unvoiced speech.

7. CONCLUSIONS

The ICRRA offers reduced reconstruction distortion in various quantization tasks compared to the GLA as a result of the higher quality of the designed codebook. A perceptually motivated spectral distortion was used to measure performance of the ICRRA. Because of the nature of the data (inverse filtered glottal pulses), distortion levels in dB similar to those typically obtained when coding the speech waveform cannot be expected. Therefore, although numerically small due to the chosen metric, the ICRRA yielded 7.2% smaller distortion compared to the GLA (experiment 2). The improvement of the ICRRA over the GLA was 48% of the improvement of the GLA over random selection of codebook vectors. The improvement has very high statistical confidence. When examining speech synthesized using the quantized source waveform, the ICRRA yielded a SNRseg improvement of several tenths (~0.2-0.3) of a dB over GLA.

It is evident from the analysis that the cost of additional storage space required by the ICRRA is negligible. In terms of increased computation, the cost is considerable. However, it can be traded-off with the introduced quantization distortion making the ICRRA particularly suitable to off-line quantization tasks such as data compression.

8. REFERENCES