World Applied Sciences Journal 10 (Special Issue of Computer & Electrical Engineering): 41-48, 2010
ISSN 1818-4952

Enhancement of LPC-10 Speech Coder Using LSP Parameters and Neural Vector Quantizers

[1]Mansour Sheikhan, [2]Davood Gharavian and [1]Ali Eslamzadeh

[1]Department of Electrical Engineering Islamic Azad University, South Tehran Branch, Tehran, Iran
[2]Department of Electrical Engineering Shahid Abbaspour University, Tehran, Iran

Abstract: High quality speech with low bit rate is assumed as an important target in the voice telecommunications systems. In this way, Linear Predictive Coder (LPC) parameters are widely used in speech coding algorithms. On the other hand, Line Spectral Pairs (LSPs) result in useful properties for quantization. In addition, Artificial Neural Networks (ANNs) have been widely used to improve the quality of speech coders and reduce the computational complexity of speech coding algorithms. In this paper, the bit rate of standard LPC-10 coder is reduced by applying two modifications in its algorithm: 1) using LSP parameters instead of LPC parameters, 2) employing fuzzy Adaptive Resonance Theory Mapping (ARTMAP) neural network and also a modified version of Kohonen neural network, which is supervised, as neural vector quantizers. Empirical results show that the bit rate of coder is reduced to 1.9 kbps and the quality of synthesized speech, in terms of Mean Opinion Score (MOS), is improved. The execution time of algorithm is reduced significantly, too.

Key words: Speech coding · LSP parameters · Neural network · Vector quantizer

INTRODUCTION

Most of the very low bit rate speech coders are based on the Linear Predictive Coding (LPC). In this algorithm, the short-term spectrum of speech is modeled by an all-pole filter [1]. However, the LPC parameters are not very efficient for quantization [2, 3] because they have no bound and so the quantization region definition is difficult. In this paper, Line Spectral Pairs (LSPs) are used as more efficient parameters [4]. On the other hand, Artificial Neural Networks (ANNs) have been used extensively and successfully for a variety of applications in speech and language technology (e.g., in speech synthesis [5-7], Automatic Speech Recognition (ASR) [8-11] and Natural Language Processing (NLP) [12] that have been experienced by the authors for Farsi language). The researches on using ANNs in speech coding can be classified into two main domains: prediction by neural models that improve the quality of coders [13-18] and reduction the computational complexity of coding algorithms [19-26].

In this paper, two modifications are proposed to apply in the structure of standard LPC-10 speech coder [27] to reduce its 2.4 kbps bit rate: 1) using LSP parameters instead of LPC parameters, 2) employing a modified version of Kohonen neural network with supervised training algorithm and also a fuzzy Adaptive Resonance Theory Mapping (ARTMAP) neural network as neural vector quantizers to reduce the computational complexity of coding algorithms.

The remainder of paper is structured as follows. Section 2 gives an overview of the standard LPC-10 coding algorithm. In section 3, the conversion of LPC to LSP parameters is reviewed. Section 4 gives the details of vector quantization by fuzzy ARTMAP and modified Kohonen neural networks. The proposed model is introduced in section 5 and the experimental results are reported in section 6. Conclusions are also discussed in section 7.

Standard LPC-10 Algorithm: The LPC-10 speech coder is the US standard for linear predictive coding of speech at 2400 bits per second. This algorithm uses

Corresponding Author: Mansour Sheikhan, Department of Electrical Engineering, Islamic Azad University, South Tehran Branch, Tehran, Iran. Tel: +98 21 88605565, Fax: +98 21 33717140, E-mail: msheikhn@azad.ac.ir.
Postal Address: Dr. Mansour Sheikhan, P.O. Box 11365/4435, Post-Graduate Center, South Tehran Branch, Islamic Azad University, Tehran, Iran

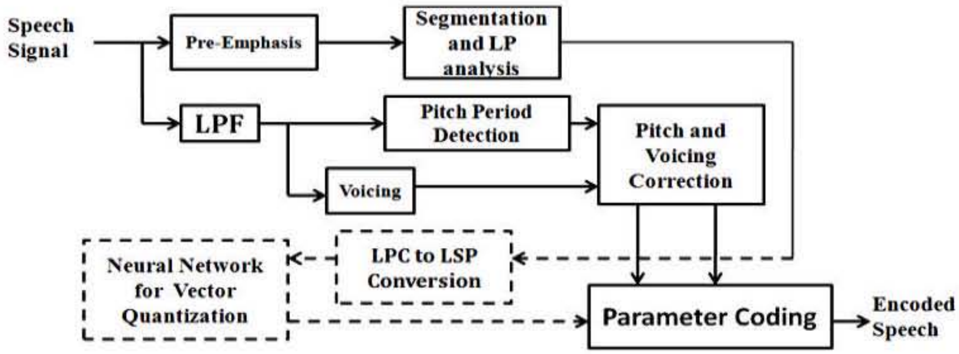

Fig. 1: Block diagram of the standard LPC-10 speech coder with the proposed modifications (shown with dashed boxes)

the analysis-by-synthesis technique, which is based on the 10th order lattice filter, to achieve the prediction parameters. The frame length in this coder is 22.5 ms and 54 bits are needed to encode the parameters in each frame. The block diagram of LPC-10 coder and also the proposed modifications in its structure, showed by dashed boxes, are depicted in Figure 1.

In this coder, speech signal is pre-emphasized by means of a first-order Finite Impulse Response (FIR) high-pass filter. The segmentation and frame processing are depending on the voicing information. The information of voicing and pitch are extracted from the output signal of a low-pass filter with a cutoff frequency of 800 Hz. Average Magnitude Difference Function (AMDF) is used to extract the pitch of filtered waveform [28]. Voiced/Unvoiced (V/UV) decision is based on the energy and Zero-Crossing Rate (ZCR) measurements and the ratio of AMDF(max) to AMDF(min) is calculated and used for this purpose.

Conversion of LPC to LSP Parameters: LSP is a mathematical model which represents the natural resonant frequencies of the human vocal tract [4]. For this purpose, assume the transfer function of vocal tract, H(z), as follows:

$$H(z) = \frac{1}{A(z)} = \frac{1}{1 - \sum_{k=1}^{p} a_k z^{-k}} \quad (1)$$

in which H(z) represents an all-pole digital filter. However, the general transfer function of a real vocal tract has both poles and zeros. So, in LSP technique A(z) is mapped into other equivalent polynomials to represent the poles and zeros in the mentioned transfer function. In this way, A(z) is decomposed into a symmetric and an asymmetric polynomial by adding and subtracting the time reversed system function as follow:

$$P(z) = A(z) + z^{-(k+1)} A(z^{-1}) \quad (2)$$

$$Q(z) = A(z) - z^{-(k+1)} A(z^{-1}) \quad (3)$$

The roots of the above mentioned polynomials are defined as the LSPs.

Neural Vector Quantization: Based on the similarity of patterns, Vector Quantization (VQ) techniques classify the k-dimensional input patterns of vector space. Each input pattern group in this classification is then represented by a vector that is called codevector or codeword. The advantage of using neural-VQ techniques in the LPC-10 speech coding algorithm is exploited in this work by quantizing the LSP parameters. In this way, two neural vector quantizer models are investigated to reduce the bit rate of the standard algorithm: 1) fuzzy ARTMAP, 2) Kohonen network with a supervised training algorithm.

Fuzzy ARTMAP Neural Network: The Fuzzy ARTMAP Neural Network (FAMNN) has been introduced by Carpenter *et al.* in 1992 [29]. FAMNN has been successfully applied in many tasks such as data mining, remote sensing and pattern recognition. FAMNN is considered as a fast trainable neural network among the members of ARTMAP family due to the computationally "cheap" mapping between inputs and outputs. Furthermore, FAMNN requires less memory since it uses a compressed representation of the data as compared to the standard nearest neighbor technique. Also the classification time of FAMNN is low. The FAMNN is a supervised network which is composed of two Fuzzy ART modules, ART_a and ART_b, as shown in Figure 2. The mentioned modules are linked together via an inter-ART module F^{ab} that is called map field. The map field is used to realize the match tracking rule, whereby the vigilance

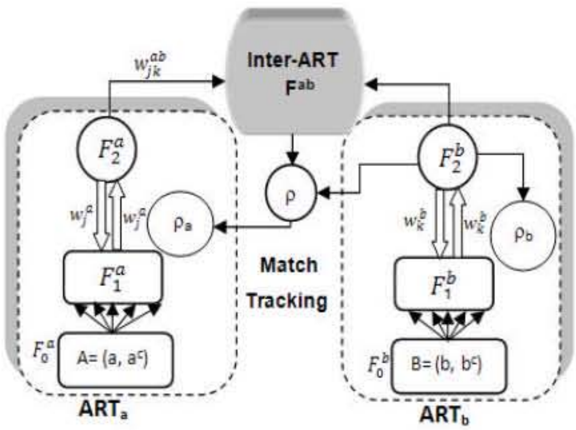

Fig. 2: Fuzzy ARTMAP structure [23]

parameter of ART_a increases in response to a predictive mismatch at ART_b. Match tracking reorganizes category structure so that predictive error is not repeated on subsequent presentations of the input.

As shown in Figure 2, variables in ART_a and ART_b modules are showed by subscripts or superscripts a and b. The inputs to each module are in the complement code form, $A = (a,a^c)$ for ART_a module and $B = (b,b^c)$ for ART_b module, respectively. For ART_a module, $x^a = \left(x_1^a,...,x_{2Ma}^a\right)$ represents the F_1^a output vector, $y^a = \left(y_1^a,...,y_{2Ma}^a\right)$ represents the F_2^a output vector and $w_j^a = \left(w_{j1}^a,...,w_{j2Ma}^a\right)$ represents the j^{th} weight vector. For ART_b module, $x^b = \left(x_1^b,...,x_{2Mb}^b\right)$ represents the F_1^b output vector and $y^b = \left(y_1^b,...,y_{2Mb}^b\right)$ represents the F_2^b output vector and $w_k^b = \left(w_{k1}^b,...,w_{k2Mb}^b\right)$ represents the k^{th} weight vector. For the map field, $x^{ab} = \left(x_1^{ab},...,x_{2Nb}^{ab}\right)$ represents the F^{ab} output vector and $w_j^{ab} = \left(w_{j1}^{ab},...,w_{j2Nb}^{ab}\right)$ represents the weight vector from j^{th} F_2^a node to F^{ab}. The map field F^{ab} is activated whenever one of the ART_a or ART_b categories is active. If node J of F^{ab} is chosen, then its weights w_J^{ab} activate F^{ab}. If node K of F_2^b is active, then the node K in F^{ab} is activated by 1-1 pathways between F_2^b and F^{ab}. The F^{ab} output vector x^{ab} obeys the following relation:

$$x^{ab} = \begin{cases} y^b \wedge w_J^{ab}; & J^{th} \text{ node of } F_2^a \text{ is activated and } F_2^b \text{ is activated} \\ w_J^{ab}; & J^{th} \text{ node of } F_2^a \text{ is activated and } F_2^b \text{ is inactivated} \\ y^b; & J^{th} \text{ is inactivated and } F_2^b \text{ is activated} \\ 0; & F_2^a \text{ and } F_2^b \text{ are inactivated} \end{cases} \quad (4)$$

The fuzzy ARTMAP module vectors [30] are based on two separately distance criteria, match and choice. The match function is defined as follows:

$$S_j(I) \equiv \frac{|I \wedge w_J|}{|I|} \quad (5)$$

Where w_j is an analog-valued weight vector associated with cluster j, $\wedge$ denotes the fuzzy AND operator, $(p \wedge q)i = min\ (p_i, q_i)$ and the norm|.|is defined by $|p| \equiv \sum_{i=1}^{M} |p_i|$.

The choice function is defined as follows:

$$T_j(I) = \frac{|I \wedge w_j|}{\alpha + |w_j|} \quad (6)$$

Where α is a small constant. Increasing α biases the search more towards the modules with large w_j. The input vector (I) is assigned to the category which maximizes $T_j(I)$ while satisfying $S_j(I) \geq \rho$, where the vigilance ρ is a constant, $0 \leq \rho \leq 1$. The fuzzy ARTMAP learning rule is given as follows:

$$w_{Ji}^{new} = \begin{cases} w_{Ji}^{old} & ; w_{Ji} \leq I_i \\ w_{Ji}^{old} - \beta(w_{Ji}^{old} - I_i) & ; w_{Ji} > I_i \end{cases} \quad (7)$$

Where $0 < \beta \leq 1$. In this way, only the weights of that cluster to which I has been assigned are updated. All w_{ij} are initially set to 1.

The map field is fundamentally a look-up table, retrieving an analog-valued weight w_{JL}^{ab} when node J of module a and node L of module b are active. Note that only one node of each module is active at a given time. If $w_{JL}^{ab} < \rho^{ab}$, then the vigilance of module a, ρ^a, is increased until node J becomes inactive. This process is repeated until $w_{JL}^{ab} \geq \rho^{ab}$. When the next input is presented, ρ^a is returned to its baseline value. All w_{JL}^{ab} are initially set to 1. During learning, when nodes J and L become active and the $w_{JL}^{ab} \geq \rho^{ab}$ inequality is valid, then all w_{JL}^{ab}; $l \neq L$, are reduced (typically set to 0).

Supervised Kohonen Network Algorithm: Kohonen neural network is typically trained in an unsupervised fashion. Some attempts were made to utilize teacher signals, if available, with the goal to improve the mapping precision. Supervised training of Kohonen network is first attempted in [31]. The supervision in [31] is achieved by attaching the information of class membership to the input vector. It is noted that during the recognition phase, the class label is omitted.

In detail, the method works as follows [32]: Assume a self organizing map M with k neurons. Each neuron in this map is associated with a codebook entry $m \in R_n$. The best matching neuron m_r for an input x is obtained by using the Euclidean distance:

$$r = \arg\min_i \|(x - m_i)\Lambda\| \tag{8}$$

Where Λ is an $n \times n$ diagonal matrix. The diagonal elements $\lambda_{11}, \ldots, \lambda_{pp}$ of this matrix are set to μ_1 and all of the remaining elements are set to μ_2. The values μ_1 and μ_2 weight the influence of components l and c in x. Then the i^{th} element of j^{th} codebook vector m_j is updated as follows:

$$\Delta m_{ij} = \begin{cases} -\beta\alpha(t) f(\Delta_{jr}) h(x_i, m_{ij}); & x \text{ and } m_r \text{ are in different classes.} \\ \alpha(t) f(\Delta_{jr})(x_i - m_{ij}); & \text{else} \end{cases} \tag{9}$$

$f(\Delta_{jr})$ in (9) is a neighborhood function which will be explained later, $\alpha(t)$ is the learning rate which is decreased linearly to zero. β is a rejection rate which weights the influence of rejection term $h(.)$. The purpose of using rejection term is to move m_r and its neighbors away from x. The effect shows itself as a reduction of the likelihood that an input node activates a codebook vector which is assigned to a foreign class in subsequent iterations. The rejection term is defined as follows:

$$h(x_i, m_{ij}) = \mathrm{sgn}(x_i - m_{ij})(\rho_i - |x_i - m_{ij}|) \tag{10}$$

Where sgn(.) is the sign function returning the sign of its argument and ρ_i is the standard deviation defined as follows:

$$\rho_i = \sqrt{\frac{\sum_{l=1}^{N} (xli - \bar{x}_i)^2}{N}} \quad ; \quad \bar{x}_i = \frac{\sum_{l=1}^{N} x_{li} \cdot \rho_i}{N} \tag{11}$$

Where N is the number of nodes in the training set. The rejection term dictates stronger actions if a codebook entry is very similar to the input node. The neighborhood function $f(.)$ can take the form of a Gaussian function as follows:

$$f(\Delta_{ir}) = \exp\left(-\frac{\|l_i - l_r\|^2}{2\sigma(t)^2}\right) \tag{12}$$

Where $\sigma(t)$ is the spread and is decreased with the number of iterations and l_r is the location of winning neuron and l_i is the location of i^{th} neuron in the lattice. It is noted that based on (9), the weights will not be distributed as a linear function of the input density. A way to obtain the pair of weight values is suggested in [32]:

$$\frac{\mu_1}{\mu_2} = \frac{n}{m} \frac{\sum_{j=1}^{m} (\phi(|l_j|) - \sigma(|l_j|))}{\sum_{i=1}^{n} (\phi(c_i) - \sigma(c_i))} \tag{13}$$

Where $\phi(l_i)$ is the average absolute value of i^{th} element of all data labels in the data set. Similarly, $\phi(c_i)$ is the average of i^{th} element of all coordinates. To obtain unique value pairs, we make the assumption that $\mu_1 + \mu_2 = 1$.

Proposed Model: In this paper, two modifications are applied to the standard LPC-10 algorithm: 1) Using LSP parameters instead of LPC parameters, 2) Reducing the bit rate of parameters in voiced and unvoiced frames using two models of neural vector quantizers.

In this way, the new model of LPC-10 coder contains of approximately 36 frames per second. It is noted that the frame length of original LPC-10 coder was 22.5 ms and contained approximately 44 frames per second. The proposed arrangement of bits for each parameter in voiced and unvoiced frames are listed in Table 1.

As shown in Table 1, the bit rate of P_1 to P_4 of this model is less than the standard LPC-10 in the voiced and unvoiced frames. It used the total of 20 bits for P_1 to P_4 parameters in the standard LPC-10, but the proposed model uses 16 bits. However, the P_9 and P_{10} of the new model use more bits than the original LPC-10 with the aim of better quality in high frequencies. Comparison of the Pulse Code Modulation (PCM) technique and the original LPC-10 coder shows that LPC-10 can not completely cover the formants in high frequency ranges (2.5-4 kHz). In contrast, the proposed model which is based on LSP parameters can keep the formants in the high frequency range better than the original LPC-10. Also it uses only 7 bits for the error protection in the unvoiced frames, because of LSP's more efficiency than LPC in frequency domain.

Table 1: Proposed arrangement of bits per frame in the standard and proposed LPC-10 speech coders

	Number of bits per frame			
	Standard Coder		Proposed Coder	
Parameters	Voiced	Unvoiced	Voiced	Unvoiced
Pitch Period	7	7	7	7
Gain	5	5	5	5
P_1	5	5	8	8
P_2	5	5		
P_3	5	5	8	8
P_4	5	5		
P_5	4	-	4	-
P_6	4	-	4	-
P_7	4	-	4	-
P_8	4	-	4	-
P_9	3	-	4	-
P_{10}	2	-	4	-
Synchronization	1	1	1	1
Error Protection	-	21	-	7
Total	54	54	53	36

Kohonen neural vector quantizer has 256 neurons in the output layer and each of [P_1,P_2] or [P_3,P_4] parameters are the inputs of this network for voiced or unvoiced frames. The same inputs are used in the fuzzy ARTMAP model.

Experimental Results: In this study, the speech database uses some of the Farsi speech data files of FARSDAT [33]. FARSDAT is a continuous speech Farsi corpus including 6000 utterances from 300 speakers with various accents. In this work, training dataset includes 160 utterances of FARSDAT sentences.

We used a learning rate $\alpha(0)$=0.7, a rejection rate $\beta(0)$=0.1, a neighborhood spread σ=40, μ_1=0.35 and μ_2=0.65 in simulations. The fine tuning of Kohonen neural network is also performed, in which learning rate is set 0.05. α=β=0.01 and ρ=0.95 are used in the simulations of fuzzy ARTMAP neural model. As a sample case, the utterance of an emotional Farsi sentence and the synthesized speech by original LPC-10 and the two mentioned proposed models are depicted in Figure 3.

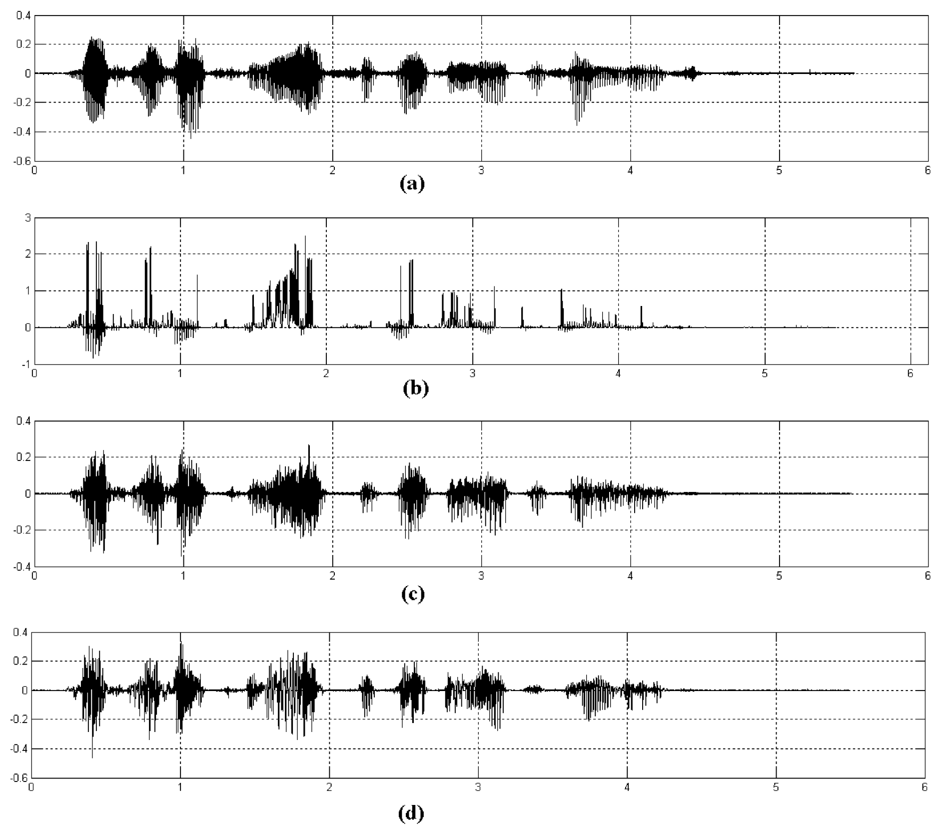

Fig. 3: Comparison of synthesized speech by different coders, a) original speech, b) LPC-10 coder, c) proposed model with Fuzzy ARTMAP-VQ, d) proposed model with supervised Kohonen self-organizing feature map (KSOFM)-VQ

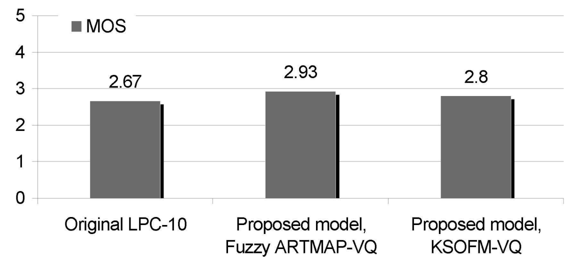

Fig. 4: MOS levels of the original LPC-10 and proposed models

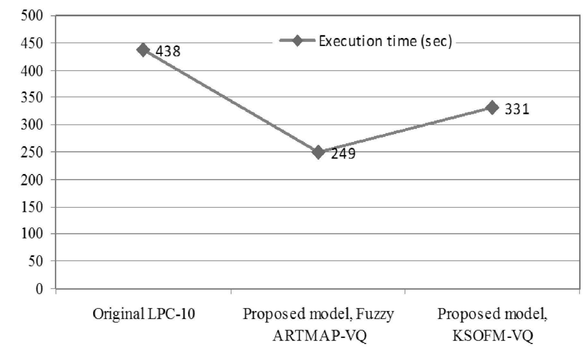

Fig. 5: Execution time for processing 18,370 speech frames in the original LPC-10 and proposed models

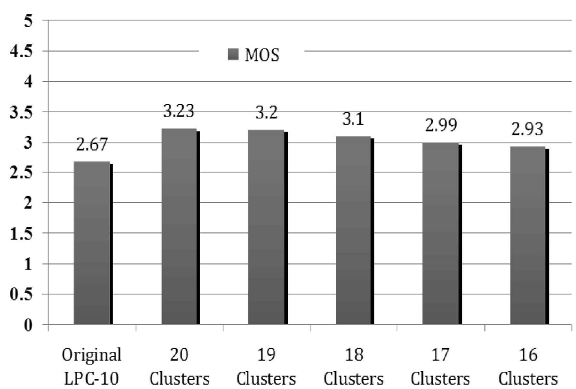

Fig. 6: MOS levels for different number of cluster groups in fuzzy ARTMAP neural quantizer

In this work, Mean Opinion Score (MOS) is used to measure the quality of synthesized speech. Thirty volunteers were the listeners in the MOS evaluation of this study. The volunteers were researchers, postgraduate and undergraduate students. The quality of synthesized speech of 40 uttered sentences by original LPC-10 and the mentioned proposed models, equipped with modified Kohonen and fuzzy ARTMAP neural networks as neural quantizers, is depicted in Figure 4. As shown in Figure 4, the proposed models have higher quality than the original LPC-10.

The execution times of the mentioned algorithms for processing 18,370 frames of speech are reported in Figure 5. As shown in Figure 5, the execution times of the proposed models as compared to the original LPC-10 are reduced significantly. The bit rate of the proposed coder has been reached to 1.9 kbps, measured over the mentioned 18370 frames, as well.

The MOS levels when using fuzzy ARTMAP neural vector quantizer with different number of cluster groups are also shown in Figure 6. As shown in Figure 6, the speech quality of proposed model with fuzzy ARTMAP neural vector quantizer can be improved gradually by increasing the number of cluster groups.

CONCLUSION

In this paper, two modifications have been applied to the standard LPC-10 speech coding algorithm to reduce the bit rate of this coder, while maintaining the quality of synthesized speech and also possible reduction of the computational complexity as compared to original LPC-10.

In this way, LSP parameters have been used instead of LPC parameters. In addition, neural vector quantizers, based on a modified version of Kohonen network with supervised training algorithm and fuzzy ARTMAP, have been employed to reduce the bit rate.

Experimental results have shown that by using the modified version of Kohonen network and fuzzy ARTMAP neural vector quantizer models, the quality of synthesized speech is improved by 0.13 and 0.26 (in terms of MOS scale), respectively. The execution times of the proposed algorithms are also reduced by 27% and 43% as compared to the original LPC-10, respectively.

REFERENCES

1. Kondoz, A.M., 2004. Digital Speech: Coding for Low Bit Rate Communications Systems, Second Edition, John Wiley and Sons.
2. Hagen, R. and P. Hedelin, 1990. Low Bit-Rate Spectral Coding in CELP, a New LSP-Method. In the Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing, pp: 189-192.
3. Laroia, R., N. Phamdo and N. Farvardin, 1991. Robust and Efficient Quantization of Speech LSP Parameters Using Structured Vector Quantizers. In the Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing, pp: 641-644.

4. Hasegawa-Johnson, M., 2000. Line Spectral Frequencies are Poles and Zeros of the Glottal Driving-Point Impedance of a Discrete Matched-Impedance Vocal Tract Model. J. Acoustic Society of America, 108: 457-460.
5. Sheikhan, M., 2003. Prosody Generation in Farsi Language. In the Proceedings of International Symposium on Telecommunications, pp: 250-253.
6. Sheikhan, M., M. Nasirzadeh and A. Daftarian, 2005. Design and Implementation of Farsi Text to Speech System. J. Engineering Faculty, Ferdowsi University of Meshed, 17: 31-48. (in Farsi).
7. Gharavian, D. and S.M. Ahadi, 2005. The Effect of Emotion on Farsi Speech Parameters: A Statistical Evaluation. In the Proceedings of 10th International Conference on Speech and Computer, pp: 463-466.
8. Sheikhan, M., M. Tebyani and M. Lotfizad, 1997. Continuous Speech Recognition and Syntactic Processing in Iranian Farsi Language. International J. Speech Technol., 1: 135-141.
9. Sheikhan, M., 2003. Suboptimum Extracted Features and Classifier for Speaker-Independent Farsi Digit Recognizer. In the Proceedings of International Symposium on Telecommunications, pp: 246-249.
10. Gharavian, D. and S.M. Ahadi, 2006. Recognition of Emotional Speech and Speech Emotion in Farsi. In the Proceedings of International Symposium on Chinese Spoken Language Processing, 2: 299-308.
11. Gharavian, D., M. Sheikhan and M. Janipour, 2010. Pitch in Emotional Speech and Emotional Speech Recognition Using Pitch Frequency. Majlesi J. Electrical Engineering 4(1) (http:// ee.majlesi.info/ index.php/ ee/ issue/view/21).
12. Sheikhan, M., M. Tebyani and M. Lotfizad, 1996. Using Symbolic and Connectionist Approaches to Automate Editing Persian Sentences Syntacticly. In the Proceedings of International Conference on Intelligent and Cognitive Systems, pp: 250-253.
13. Birgmeier, M., 1996. Nonlinear Prediction of Speech Signals Using Radial Basis Function Networks. In the Proceedings of European Signal Processing Conference, 1: 459-462.
14. Kumar, A. and A. Gersho, 1997. LD-CELP Speech Coding with Nonlinear Prediction. IEEE Signal Processing Letters, 4: 89-91.
15. Ma, N. and G. Wei, 1998. Speech Coding with Nonlinear Local Prediction Model. In the Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing, 2: 1101-1104.
16. Faúndez-Zanuy, M., S. McLaughlin, A. Esposito, A. Hussain, J. Schoentgen, G. Kubin, W.B. Kleijn and P. Maragos, 2002. Nonlinear Speech Processing: Overview and Applications. Control and Intelligent Systems, 30: 1-10.
17. Faúndez-Zanuy, M., 2003. Nonlinear Speech Coding with MLP, RBF and Elman Based Prediction. Lecture Notes in Computer Sci., 2687: 671-678.
18. Sheikhan, M., V. Tabataba Vakili and S. Garoucy, 2009. Complexity Reduction of LD-CELP Speech Coding in Prediction of Gain Using Neural Networks. World Applied Sciences J. 7(Special Issue of Computer and IT): 38-44.
19. Easton, M.G. and C.C. Goodyear, 1991. A CELP Codebook and Search Technique Using a Hopfield Net. In the Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing, pp: 685-688.
20. Hernandez-Gomez, L.A. and E. Lopez-Gonzalo, 1993. Phonetically-Driven CELP Coding Using Self-Organizing Maps. In the Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing, 2: 628-631.
21. Wu, L., M. Niranjan and F. Fallside, 1994. Fully Vector-Quantized Neural Network-Based Code Excited Nonlinear Predictive Speech Coding. IEEE Transactions on Speech and Audio Processing, 2: 482-489.
22. Wu, S., G. Zhang, X. Zhang and Q. Zhao, 2008. A LD-CELP Speech Coding Algorithm Based on Modified SOFM Vector Quantizer. In the Proceedings of International Symposium on Intelligent Information Technol. Application, pp: 408-411.
23. Sheikhan, M., V. Tabataba Vakili and S. Garoucy, 2009. Codebook Search in LD-CELP Speech Coding Algorithm Based on Multi-SOM Structure. World Applied Sciences J., 7(Special Issue of Computer and IT): 59-68.
24. Sheikhan, M. and S. Garoucy, 2010. Reducing the Codebook Search Time in G.728 Speech Coder Using Fuzzy ARTMAP Neural Networks. World Appl. Sci. J., 8(10): 1260-1266.
25. Sheikhan, M., V. Tabataba Vakili and S. Garoucy, 2010. A New 12.8 kbps LD-CELP Speech Codec Based on a Neural Search Codebook Module. Majlesi J. Electrical Engineering, Article in Press.
26. Sheikhan, M. and S. Garoucy, 2010. Hybrid VQ and Neural Models for ISF Quantization in Wideband Speech Coding. World Applied Sciences J. Article in Press.

27. Tremain, T., 1982. The Government Standard Linear Predictive Coding Algorithm: LPC-10. Speech Technol., 1: 40-49.
28. Spanias, A., 1994. Speech Coding: A Tutorial Review. Proceedings of the IEEE, 82: 1541- 1582.
29. Carpenter, G.A., S. Grossberg, N. Markuzon, J.H. Reynolds and D.B. Rosen, 1992. Fuzzy ARTMAP: A Neural Network Architecture for Incremental Supervised Learning of Analog Multidimensional Maps. IEEE Transactions on Neural Networks, 3: 698-713.
30. Matthias, B. and C.E. Sadik, 1995. Optoelectronic Fuzzy ARTMAP Processor. J. Optical Computing, 10: 213-215.
31. Kohonen, T., 1995. Self-Organizing Maps, Springer Series in Information Sciences.
32. Hagenbuchner, M., A. Sperduti and A. Tsoi, 2003. A Self-Organizing Map for Adaptive Processing of Structured Data. IEEE Transactions on Neural Networks, 14: 491-505.
33. Bijankhan, M., J. Sheikhzadegan, M.R. Roohani, Y. Samareh, C. Lucas and M. Tebiani, 1994. The Speech Database of Farsi Spoken Language. In the Proceedings of the Fifth Australian International Conference on Speech Science and Technology (SST'94), pp: 826-831.