World Applied Sciences Journal 7 (Special Issue of Computer & IT): 38-44, 2009
ISSN 1818-4952

Complexity Reduction of LD-CELP Speech Coding in Prediction of Gain Using Neural Networks

[1]*Mansour Sheikhan,* [2]*Vahid Tabataba Vakili and* [1]*Sahar Garoucy*

[1]Department of Electrical Engineering, Islamic Azad University, South Tehran Branch, Tehran, Iran
[2]Department of Electrical Engineering, Iran University of Science and Technology, Tehran, Iran

Abstract: Reducing the computational complexity is desired in speech coding algorithms. In this paper, three neural gain predictors are proposed which can function as backward gain adaptation module of low delay-code excited linear prediction (LD-CELP) G.728 encoder, recommended by International Telecommunication Union-Telecom sector (ITU-T, formerly CCITT). Elman, multilayer perceptron (MLP) and fuzzy ARTMAP are candidate neural models in this work. Empirical results show that gain prediction by Elman and MLP neural networks improve the mean opinion score (MOS) and segmental signal to noise ratio (SNR_{seg}) as compared to traditional implementation of encoder. However, fuzzy ARTMAP reduces the computational complexity noticeably, without significant degradations in MOS and SNR_{seg}.

Key words: Gain prediction . LD-CELP speech encoder . neural networks

INTRODUCTION

With the fast progress of communication systems and penetration rate increase of mobile and Internet, employing speech compression algorithms for efficient utilization of channel bandwidth is necessary. So far, various methods for speech coding have been proposed. In this way, one of the most effective methods is based on the analysis by synthesis (AbS) technique, which was established by Atal and Remede in 1982 [1]. One of the main algorithms for speech coding at bit rates lower than 16 kbps is code excited linear prediction (CELP). It was introduced by Schroder and Atal in 1985 [2]. In May 1992, International Telecommunication Union-Telecom sector (ITU-T, formerly CCITT) approved a 16 kbps low delay CELP (LD-CELP) coding algorithm with a delay of less than 2 msec and recommended it as G.728 [3]. In 1994, fixed-point version of LD-CELP was introduced [4]. LD-CELP is basically a backward-adaptive version of the CELP coder in which the predictor and excitation gain are updated backward adaptively by analyzing the former quantized speech and excitation, respectively. This coder is generally used in Internet voice calls and cell phones. Many researches have been performed to improve LD-CELP speech coding algorithm [5-11].

On the other hand, artificial neural networks (ANNs) emerged in the recent decades as powerful and adaptive data processing models for pattern classification and feature extraction. Neural networks have been used extensively and successfully for a variety of applications in speech coding algorithms, as well. The researches on using ANNs in speech coding can be classified into two main domains: neural predictors which improve the quality of coder [12-20] and reduction the computational complexity [21-26].

CELPs with different nonlinear predictors were proposed to improve the signal to noise ratio (SNR) of the decoded signal [14-19]. For example, a nonlinear scalar predictor based on hybrid of three ANNs is introduced in [18]. In the family of CELP coders, codebook search process has high complexity. ANNs can be used to reduce this complexity. For example, an efficient procedure for exploiting self organizing maps (SOMs) for a fast search quantization procedure is presented in [21] that greatly reduce the complexity for vector quantization (VQ) of the spectral envelope. Huong *et al*. employed a new line spectral pairs (LSPs) codebook by using a centroid neural network (CNN) to enhance the compression rate of an adaptive multi-rate (AMR) coder [22]. Stochastic codebook (SCB) search for CELP coding is performed by counter propagation neural network model and less computational complexity is achieved [23]. A modified Hopfield neural net is also used to search in codebook of a CELP coder [24]. A codebook design algorithm, based on modified self-organizing feature map (SOFM) neural network, is introduced for LD-CELP in [25], as well.

In the LD-CELP coder, there are three separate LPC analyses to update the coefficients of three filters:

Corresponding Author: Dr. Mansour Sheikhan, P.O. Box 11365/4435,
Post-Graduate Center, South Tehran Branch, Islamic Azad University, Iran

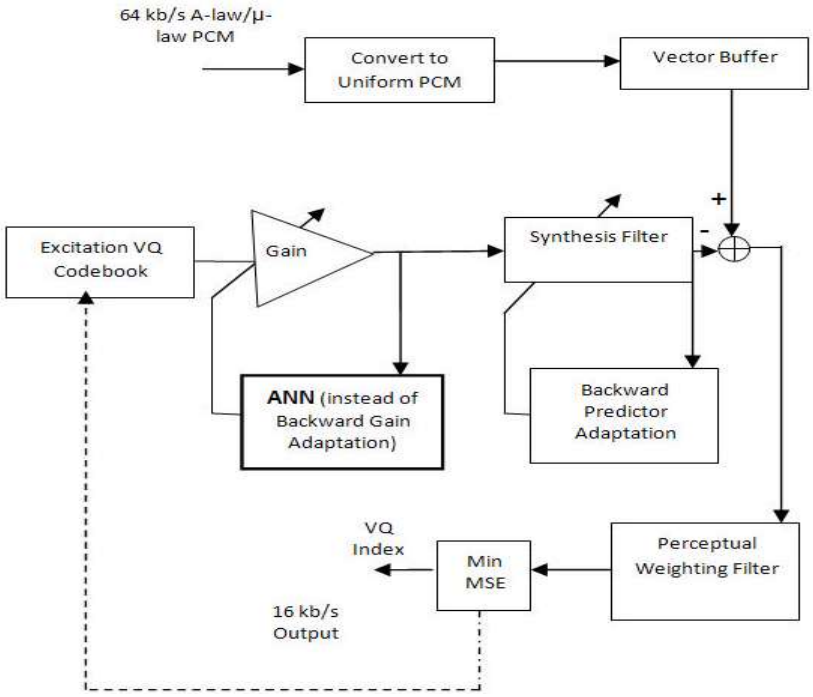

Fig. 1: Block diagram of LD-CELP encoder and the proposed modification

the synthesis filter, the log-gain predictor and the perceptual weighting filter. This paper is focused on the log-gain predictor. In this study, gain prediction is preformed by using neural networks to reduce the complexity of LD-CELP algorithm. Three neural models are used in this paper to predict gain: Elman, multilayer perceptron (MLP) and fuzzy ARTMAP.

It is noted that Elman neural model is commonly a two-layer network with feedback from output of the first-layer to the input. This recurrent connection allows the Elman NN to both detect and generate time-varying patterns [27, 28]. MLP is another candidate neural model which is an excellent function approximation tool, utilizing hidden units and have been employed with great success in both classification and regression analysis problems [29, 30]. Fuzzy ARTMAP, which is an extension to ARTMAP and it is capable of forming associative maps between clusters of its input and output domains in a supervised manner [31-35], is another neural model that is used in the experiments of this research.

This paper is organized as follows. Section 2 gives a brief overview of the 16 kbps LD-CELP coder. In Section 3, backward adaptation of excitation gain in LD-CELP is described. The details of gain prediction by Elman, MLP and fuzzy ARTMAP neural models are discussed in Section 4. In Section 5, the experimental results are reported and conclusions are drawn in Section 6.

ARCHITECTURE OF LD-CELP CODER

LD-CELP is an AbS codebook driven method for linear predictive speech coding [3, 7]. The basic structure of the encoder and the proposed modification in its block diagram are shown in Fig. 1. In this coder, which is an encoding method based on a source filter model, speech is reproduced by using excitation codevectors. These codevectors are time-series signals which are stored in an excitation codebook and drive a linear predictive synthesis filter that represents the spectral envelope of the input speech.

The optimal excitation codevector is selected from the excitation codebook by using a closed-loop search according to the AbS method to find the one having the minimum perceptually-weighted waveform distortion of the synthetic speech signal.

BACKWARD ADAPTATION OF EXCITATION GAIN IN LD-CELP

In this section, backward adaptation of excitation gain module, as a part of LD-CELP algorithm, is explained (Fig. 2). Input and output of this module are gain-scaled excitation $e(n)$ and excitation gain $\sigma(n)$, respectively. As shown in Fig. 2, the 1-vector delay unit makes the previous gain-scaled excitation vector $e(n-1)$ available. The root-mean-square (RMS) calculator then calculates the RMS value of the vector $e(n-1)$. Then,

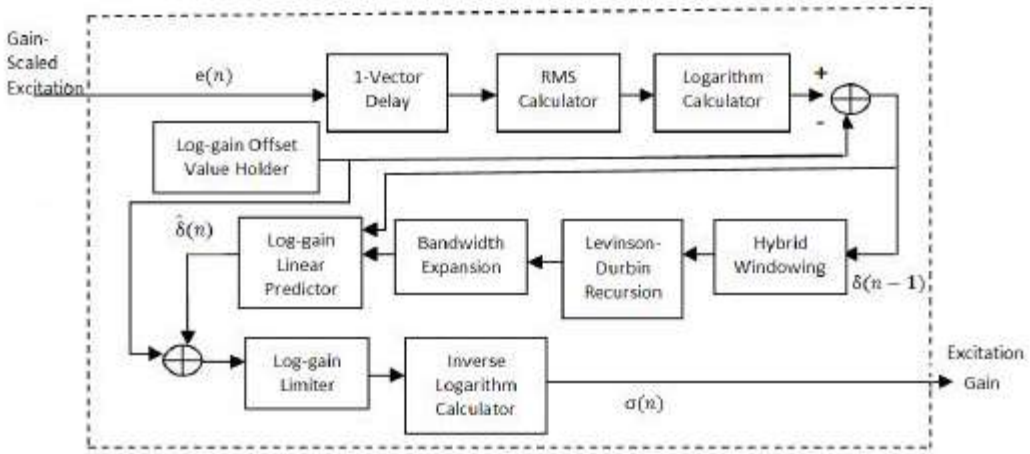

Fig. 2: Block diagram of backward gain adaptation in LD-CELP [3]

Table 1: Number of instruction cycles for backward gain adaptation module components of ITU-T G.728 [36]

Functional block	Number of instruction cycles
Linear prediction in logarithmic domain	39
Offset value adder	2
Logarithmic gain limiter	6
Inverse logarithm calculator	27
Hybrid windowing	253
Levinson-Durbin recursion	1375
Bandwidth expansion	5
Total for each codeword	1707

the logarithm calculator calculates the dB value of the RMS of *e(n–1)*. A log-gain offset value of 32 dB is stored in the log-gain offset value holder. This value is roughly equal to the average excitation gain level (in dB) during voiced speech. The adder subtracts this log-gain offset value from the logarithmic gain produced by the logarithm calculator. The resulting offset removed logarithmic gain, *δ(n–1)*, is then used by the hybrid windowing module and the Levinson-Durbin recursion module. The output of the Levinson-Durbin recursion module is the coefficients of the tenth order of linear predictor. The bandwidth expansion module then moves the roots of this polynomial toward the origin of z-plane. The predictor attempts to predict *δ(n)* based on a linear combination of *δ(n–1)*, *δ(n–2)*,..., *δ(n–10)* [6, 7]. The predicted version of *δ(n)* is denoted as $\hat{\delta}(n)$ and is given by:

$$\hat{\delta}(n) = -\sum_{i=1}^{10} a_i \delta(n-i) \qquad (1)$$

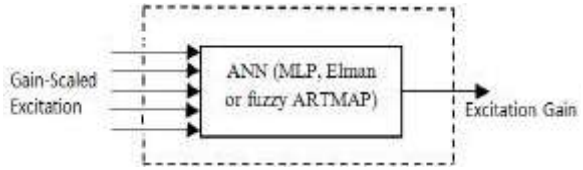

Fig. 3: Replacement of backward gain adaptation module by ANN models

In the next step, offset value adds to $\hat{\delta}(n)$ and then the log-gain limiter clips the level of it, if the resulting log-gain value was lower or upper than 0 dB and 60 dB, respectively. Finally, the value of log-gain in logarithmic domain converts to linear domain by inverse logarithm calculator.

The complexity of backward gain adaptation module components, in terms of instruction cycles, is reported in Table 1.

As shown in Fig. 2, LPC analysis is used in the structure of backward gain adaptation module to update the coefficients of filter. Levinson-Durbin algorithm and hybrid windowing are the most important factors of this complexity. Operations in both of these modules depend strongly on the order of LPC, *p*, and frame size, *L*. If the width of hybrid non-recursive window is equal to *N*, then hybrid windowing needs $(L+N)*P+1$ multiplications and $(L+N)*P+1$ additions and Levinson-Durbin algorithm needs P^2+2P multiplications and P^2 additions. In this paper, backward gain adaptation module is replaced by ANN models to reduce the complexity of algorithm (Fig. 3).

NEURAL GAIN PREDICTOR CANDIDATES

As mentioned earlier, three neural models (Elman, MLP and fuzzy ARTMAP) are used to predict

Table 2: Elman-based gain predictor specifications

Specification	Value or type
Train function	'trainlm'
Net.trainParam.goal	0.01
Number of nodes in layers	8-8-1
Activation functions of layers	'tansig', 'tansig', 'purelin'
Number of epochs	321
Training time (sec)	2200
MSE on the test data	0.019

Table 3: MLP-based gain predictor specifications

Specification	Value or type
Train function	'trainlm'
net.trainParam.goal	0.01
Number of nodes in layers	8-8-1
Activation functions of layers	'tansig', 'tansig', 'purelin'
Number of epochs	620
Training time (sec)	3100
MSE on the test data	0.029

Table 4: Fuzzy ARTMAP-based gain predictor specifications

Specification	Value
Learning rate	0.968
Vigilance parameter	0.955
Number of F_0 nodes	10
Number of F_1 nodes	540
Number of F_2 nodes	540
Number of epochs	1
Training time (sec)	301
Correct identification rate	95%

excitation gain in LD-CELP encoder. The scaled excitation vector ($e(n)$), is fed as input pattern to the network and excitation gain ($\sigma(n)$) is target output of the network. Codebook search module, searches through 1024 candidate codevectors in the excitation VQ codebook and finds index of the best codevector. Indeed, in excitation VQ codebook, the best shape codevector and the best gain value, which are extracted from codebook module, are multiplied by each other to get the quantized excitation vector $y(n)$. Then, this vector multiplies by gain and results the scaled excitation vector. The excitation gain is the output of backward gain adaptation module. The sizes of scaled excitation vector and excitation gain are 5 and 1, respectively. The training dataset includes about 50,000 vectors of fifteen male and twenty female speakers with different accents. In fact, these vectors are applied to encoder and the excitation gain and scaled excitation are calculated for each frame. This data is used as the training data of neural models. The training dataset of Elman and MLP neural networks are similar. The details of fuzzy ARTMAP training dataset are explained in the next subsections, too. The test dataset includes 9,000 vectors, as well.

Based on the minimum mean squared error (MSE) and training time, the optimal neural model is selected. MSE is calculated using Equation (2):

$$MSE = \frac{\sum_{j}^{n}\sum_{i}^{m}(d_{ij} - y_{ij})2}{N_d \times n} \quad (2)$$

where n is the number of output processing elements and N_d is the number of exemplars in dataset. y_{ij} is the network output for exemplar i at processing element j and d_{ij} is the desired output for exemplar i at processing element j.

Elman NN: Elman NN is a type of partial recurrent network with an additional feedback connection from the output of the first layer to its input layer. In our scheme, Elman has tangent sigmoid, *tansig*, neurons in its two hidden layers and pure linear, *purelin*, neurons in its output layer. The specifications of Elman-based gain predictor in our simulations, using Neural Network Toolbox of MATLAB software, are reported in Table 2.

MLP NN: Multilayer perceptrons are the most popular NN with successful pattern matching and function-approximation applications in many diverse fields. The MLP in our experiments has three layers (two hidden layers and one output layer). The specifications of MLP-based gain predictor in our simulations are reported in Table 3.

Fuzzy ARTMAP: Fuzzy ARTMAP is a neural architecture for incremental supervised learning of recognition categories and multidimensional maps in response to arbitrary sequences of analog or binary input vectors. It achieves a synthesis of fuzzy logic and adaptive resonance theory (ART) neural networks by exploiting a close formal similarity between the computations of fuzzy method and ART category choice, resonance and learning. ARTMAP networks consist of two ART_1 networks, ART_a and ART_b, bridged via inter-ART module. An ART_1 module has three layers: input layer (F_0), the comparison layer (F_1) and the recognition layer (F_2). Fuzzy ARTMAP is a natural extension to ARTMAP that uses fuzzy ART instead of ART_1 modules [31-35]. The operation of fuzzy ARTMAP is affected by two network parameters: the choice parameter, a, and the baseline vigilance

Table 5: Performance comparison of the proposed systems with traditional ITU-T G.728 implementation

System	Execution time (sec)	SNR_{seg} (dB)	MOS
Traditional G.728 [7, 10]	0.25500	18.45	3.91
G.728 with Elman-based gain predictor	0.04370	18.59	4.06
G.728 with MLP-based gain predictor	0.03870	18.51	3.93
G.728 with fuzzy ARTMAP-based gain predictor	0.00129	18.37	3.71

parameter, ρ. These parameters take values in the interval [0,1]. Both of these parameters affect the number of nodes created in the category representation layer of fuzzy ARTMAP.

The dataset which is used to train the Elman and MLP networks is not suitable for fuzzy ARTMAP and some preprocessing is needed. Fuzzy ARTMAP requires input patterns to be presented as vectors of floating point numbers in the range [0,1]. Therefore the training and test datasets need normalization or mapping the original values into this range. The value of excitation gain in ITU-T G.728 recommendation is in the range of [0 dB,60 dB]. In our simulation of fuzzy ARTMAP structure, the mentioned range is divided to 540 classes. So, the resolution of this classification is about 0.1 dB. The specifications of the fuzzy ARTMAP-based gain predictor in our simulations are reported in Table 4, too.

EMPIRICAL RESULTS

In this section, we will compare the performance of three mentioned neural models which are employed for gain prediction. In this work, a 16 kbps LD-CELP coder based on the G.728 recommendation is implemented [3]. The encoder and decoder are simulated using MATLAB v.7.2 software. The speech database in this work use Farsi speech data files of FARSDAT [37]. FARSDAT is a continuous speech Farsi corpus including 6,000 utterances from 300 speakers with various accents. 210 sentences are selected, including 90 male and 120 female utterances. The sampling frequency is 8 kHz and the frame size is 20 samples, as well. This dataset consists of about 50,000 vectors. 40,000 vectors of this dataset are used for training candidate neural networks and 9,000 vectors are used as test dataset.

The performance comparison of Elman-based and MLP-based gain predictors shows that MSE in Elman is lower than MLP. The number of epochs in Elman is lower than MLP, too. However, when the backward gain adaptation module is replaced by trained network, the execution time for MLP is lower than Elman (Table 5). The execution time, when calculated for 400 frames of speech, is 0.0387 sec for MLP and is 0.0437 sec for Elman NN. By comparing the performance of fuzzy ARTMAP-based and Elman-based neural gain predictors, we conclude that the number of epochs and training time in fuzzy ARTMAP are the lowest ones. The mentioned execution time for fuzzy ARTMAP is 0.00129 sec, which is the lowest among three approaches, too. The performance of three proposed systems, equipped with neural gain prediction, can be compared with a traditional G.728 [7, 10] in terms of segmental SNR (SNR_{seg}) and mean opinion score (MOS) (Table 5).

It is noted that MOS provides a numerical indication of the perceived quality of received media after compression and/or transmission. The MOS is expressed as a single number in the range of 1 to 5, where 1 is the lowest perceived audio quality and 5 is the highest perceived audio quality measurement.

SNR_{seg} is an important factor in determining the quality of audio data, too. This is particularly important in speech recognition technology, since it is well known that recognition performance is strongly influenced by the SNR [38]:

$$SNR = 10\log\left(\frac{\sum_{n} x^2(n)}{\sum_{n} (x^2(n) - y^2(n))^2}\right) \quad (3)$$

where $x(n)$ is the input signal to encoder and $y(n)$ is the output signal from decoder. SNR_{seg} is defined as the average of SNR measurements:

$$SNR_{seg} = \frac{1}{N_f}\sum_{m=1}^{N} SNR_m \quad (4)$$

in which, N_f is the number of frames.

CONCLUSIONS

In this paper, backward gain prediction module in the structure of LD-CELP encoder was replaced by three neural gain predictors. Elman, MLP and fuzzy ARTMAP were the candidate neural models in this work. Empirical results showed that gain prediction by Elman and MLP neural networks improved the MOS

and SNR_{seg}, as compared to traditional implementations of G.728 encoder. In this way, when Elman and MLP gain predictors were used, MOS was 0.15 and 0.02 higher than traditional G.728, respectively (Table 5).

The SNR_{seg} for Elman and MLP neural predictors was also 0.14 dB and 0.06 dB higher than traditional G.728, respectively. On the other hand, fuzzy ARTMAP-based gain predictor reduced the computational complexity noticeably, without significant degradations in MOS and SNR_{seg}. Experimental results showed that when fuzzy ARTMAP is used as the neural gain predictor, MOS and SNR_{seg} were reduced 0.2 and 0.08 dB, respectively.

REFERENCES

1. Atal, B.S. and R. Remde, 1982. A New Model of LPC Excited for Producing Natural-Sounding Speech at Low Bit Rates. In the Proceedings of the International Conference on Acoustics, Speech and Signal Processing, pp: 614-617.
2. Schroeder, M.R. and B.S. Atal, 1985. Code-Excited Linear Prediction (CELP): High Quality Speech at Very Low Bit Rates. In the Proceedings of the International Conference on Acoustics, Speech and Signal Processing, 10: 937-940.
3. ITU-T G.728 Recommendation, 1992. Coding of Speech at 16 kb/s Using Low-Delay Code Excited Linear Prediction.
4. ITU-T G.728-Annex G, 1994. 16 kbps Fixed Point Specification.
5. Chen, J.H., R.V. Cox, Y.C. Lin and N. Jayant, 1992. A Low Delay CELP Coder for CCITT 16 kb/s Speech Coding Standard. IEEE Journal on Selected Areas on Commmunication, 10: 830-847.
6. Chen, J.H. and R.V. Cox, 1993. The Creation and Evolution of LD-CELP: From Concept to Standard. Speech Communication, 12: 103-112.
7. Zahang, G., K.M. Xie and L.Y. Huangfu, 2003. Optimization Gain Codebook of LD-CELP. In the Proceedings of the International Conference on Acoustics, Speech and Signal Processing, 2: 149-152.
8. Zahang, G., K.M. Xie, X.Y. Zhang and L.Y. Huangfu, 2003. Improvement on G.728 by Normalized Shape Codebook with Exact Gain. Journal of China Institute of Communications, 24: 87-92.
9. Zahang, G., K.M. Xie and L.Y. Huangfu, 2004. Improving G.728's Hybrid Window and Excitation Gain. In the Proceeding of the Asia-Pacific IEEE Conference on Circuits and System, 1: 185-188.
10. ITU-T G.728-Implementor 's Guide, 2006. Coding of Speech at 16kb/s Using Low Delay Code Excited Linear Prediction.
11. Xueying, Z., Z.Q. Qun and M.Z. Yanys, 2008. Reducing the Complexity of LD-CELP Speech Coding Algorithm Using Direct Vector Quantization. In the Proceedings of the International Conference on Communication, Circuits and Systems, 10: 811-815.
12. Kumar, A. and A. Gersho, 1997. LD-CELP Speech Coding with Nonlinear Prediction. IEEE Signal Processing Letters, 4: 89-91.
13. Ma, N. and G. Wei, 1998. Speech Coding with Nonlinear Local Prediction Model. In the Proceedings of the International Conference on Acoustics, Speech and Signal Processing, 2: 1101-1104.
14. Thyssen, J., H. Nielsen and S.D. Hansen, 1994. Nonlinear Short-Term Prediction in Speech Coding. In the Proceedings of the International Conference on Acoustics, Speech and Signal Processing, 1: 185-188.
15. Faundez, M., 1999. Adaptive Hybrid Speech Coding with a MLP/LPC Structure. In the Proceedings of the International Work-Conference on Artificial and Natural Neural Networks, 11: 814-823.
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. Birgmeier, M., 1996. Nonlinear Prediction of Speech Signals Using Radial Basis Function Networks. In the Proceedings of the European Signal Processing Conference, 1: 459-462.
18. Faúndez-Zanuy, M., 2003. Nonlinear Speech Coding with MLP, RBF and Elman Based Prediction. Lecture Notes in Computer Science, 2687: 671-678.
19. Le, T.T. and J.S. Mason, 1994. Nonlinear Predictor for Speech Enhancement. In the Proceedings of the IEE Colloquium on Techniques for Speech Processing and their Applications, 11: 4-11.
20. Sassi, S.B., R. Braham and A. Belghith, 2001. Neural Speech Synthesis System for Arabic Language Using CELP Algorithm. In the Proceedings of the ACS/IEEE International Conference on Computer Systems and Applications, pp: 119-121.

21. Hernandez-Gomez, L.A. and E. Lopez-Gonzalo, 1993. Phonetically-Driven CELP Coding Using Self-Organizing Maps. In the Proceedings of the International Conference on Acoustics, Speech and Signal Processing, 2: 628-631.
22. Huong, V., B.J. Min, D.C. Park and D.M. Woo, 2008. A New Vocoder Based on AMR 7.4 kbit/s Mode in Speaker Dependent Coding System. In the Proceedings of the ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing, pp: 163-167.
23. Indrayanto, A., A. Langi and W. Kinsner, 1991. A Neural Network Mapper for Stochastic Codebook Parameter Encoding in Code Excited Linear Predictive Speech Processing. In the Proceedings of the IEEE Western Canada Conference on Computer, Power and Communications Systems, pp: 221-224.
24. Easton, M.G. and C.C. Goodyear, 1991. A CELP Codebook and Search Technique Using a Hopfield Net. In the Proceedings of the International Conference on Acoustics, Speech and Signal Processing, pp: 685-688.
25. 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 the International Symposium on Intelligent Information Technology Applications, pp: 408-411.
26. 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.
27. Elman, J.L., 1990. Finding Structure in Time. Cognitive Science, 14: 179-211.
28. Hertz, J., A. Krogh and R.G. Palmer, 1991. Recurrent Neural Networks: Introduction to the Theory of Neural Computations, Addison-Wesley, Chapter 7.
29. Widrow, B. and M.A. Lehr, 1990. 30 Years of Adaptive Neural Networks: Perceptron, Madaline and Backpropagation. Proceedings of the IEEE, 78: 1415-1442.
30. Buntine, W.L. and A.S. Weigend, 1994. Computing Second Derivatives in Feed-Forward Networks: A Review. IEEE Transactions on Neural Networks, 5: 480-488.
31. Carpenter, G.A., S. Grossberg and D. Rosen, 2001. Fuzzy ART: An Adaptive Resonance for Rapid, Stable Classification of Analog Patterns. In the Proceedings of the International Joint Conference of Neural Networks, pp: 411-416.
32. Carpenter, G.A., S. Grossberg, N. Markuzon, J.H. Reynolds and D. Rosen, 1992. Fuzzy ARTMAP: A Neural Network Architecture for Incremental Supervised Learning of Analog Multidimensional Maps. IEEE Transactions on Neural Networks, 3: 698-713.
33. Tan, A.H., 1997. Cascade ARTMAP Integrating Neural Computation and Symbolic Knowledge Processing. IEEE Transactions on Neural Networks, 8: 237-250.
34. Dagher, I., M. Georgiopoulos, G.L. Heileman and G. Bebis, 1999. An Ordering Algorithm for Pattern Presentation in Fuzzy ARTMAP that Tends to Improve Generalization Performance. IEEE Transactions on Neural Networks, 10: 768-778.
35. Charalampidis, D., T. Kasparis and M. Georgiopoulos, 2001. Classification of Noisy Signals Using Fuzzy ARTMAP Neural Networks. IEEE Transactions on Neural Networks, 12: 1023-1036.
36. Beritelli, F., S. Casale and A. Cavallaro, 1997. Low-Complexity Fuzzy Control of Excitation Gain in LD-CELP Speech Coding. Eletronics Letters, 33: 1846-1847.
37. 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.
38. Deller, J.R., J.H.L. Hansen and J.G. Proakis, 2000. Discrete-Time Processing of Speech Signals. IEEE Press.