Improved Self-Organizing Maps and Speech Compression

Mohamed ETTAOUIL[*], **Mohamed LAZAAR**[*]

** UFR : Scientific Computing and Computer sciences, Engineering sciences,*
Modeling and Scientific Computing Laboratory,
Faculty of Science and Technology of Fez,
University Sidi Mohammed ben Abdellah Box 2202, Fez, MOROCCO

Abstract

Recent developments of Self-Organizing Maps or Kohonen networks become more and more interesting in many fields such as: pattern recognition, clustering, speech recognition, data compression, medical diagnosis... Kohonen networks is unsupervised learning models. The results obtained by the Kohonen networks are dependent on their parameters such as the architecture of the Kohonen map, the later has a great impact on the convergence of learning methods. The selection of the architecture of Kohonen networks, associated with a given problem, is one of the most important research problems in the neural network research. In this paper, we model this problem of neural architecture in terms of a mixed-integer non linear problem with linear constraints. To solve this model of optimization for the network architectures, we propose the genetic algorithm. Also, we implemented and evaluated the proposed method and speech compression algorithms. Speech compression is the technology of converting human speech into an efficiently encoded representation that can later be decoded to produce a close approximation of the original signal. The numerical results demonstrated the effectiveness of the new model.

Keywords: *Kohonen networks, genetic algorithms, Speech compression, unsupervised training, mixed-integer non-linear programming, Mel Frequency Cepstral Coefficient (MFCC).*

1. Introduction

Artificial Neural Network (ANN) often called as Neural Network (NN). It is a computational model or mathematical model based on biological neural networks.

The Artificial Neural Networks (ANN) are a very powerful tool to deal with many applications [1][5], and they have proved their effectiveness in several research areas such as analysis and image compression[30][31], recognition of writing, speech recognition [23], speech compression[22], video compression [12][32], signal analysis, process control, robotics, and research on the Web [4] [18].

Teuvo Kohonen has introduced the very interesting concept of self-organized topological feature maps [20], which are maps that preserve the topology of a multidimensional representation within the new one- or two-dimensional array of neurons. The concept of topology has become the essential feature of the Kohonen approach in neural network research [28].

Speech compression is the technology of converting human speech into an efficiently encoded representation that can later be decoded to produce a close approximation of the original signal. Speech compression can also be viewed as a form of classification, since it assigns a template or a code word to a set of input parameters vectors drawn from large parameters set in such a way as to provide a good approximation of representation. Vector quantization is the well known method as a component algorithm for loss compression methods, and has been observed as an efficient technique for data compression (see Fig. 1).

Many loss compression methods are using the LBG algorithm for the VQ, which was developed by Linde [24]. But, the LBG algorithm is iterative and requires a considerable computation time to get optimal code vectors [14]. The quantization method using the artificial neural network is well suitable to the application that the statistical distribution of the original data changes as the time passes, since it supports the adaptive learning to data [13]. Also, the neural network has a huge parallel structure and the possibility for high-speed processing.

The key advantages of the VQ representation are:

- Reduced storage for analysis information.
- Reduced computation for determining similarity of analysis vectors.
- Discrete representation of speech sounds.

The latter algorithms detail the possible methodologies for introducing expertise that follows the learning unsupervised stage. Since the training stage is very important in the Self-Organizing Maps (SOM) performance, the selection of the architecture of Kohonen network, associated with a given problem, is one of the most important research problems in the neural network research. More precisely, the choice of neurons number and the initial weights has a great impact on the convergence of learning methods. The optimization of the

artificial neural networks architectures, particularly Kohonen networks, is a recent problem. The first techniques consists building the map in an evolutionary way: allowing, adding neurons and deleting some others.
Some methods have been proposed in the literature can be broadly classified into two categories: the first fixed a priori the size of the map in an evolutionary way [33]; the second category allows the data themselves to choice the dimension of the map. The algorithm neural gas [22] built the plot by introducing the connections in the area of data. Recently, another method is introduced to determine the network parameters, in the supervised learning [9] and in the Kohonen networks [8]. The mean propose of this work is to model this choice problem of neural architecture in terms of a mixed-integer non linear problem with linear constraints. Because of its effectiveness in solving the optimization problems, the genetic algorithm approach is used to solve this non linear problem [17]. It should be noted that a good local optimum of the obtained model permits to improve the performance of the Kohonen learning algorithm.

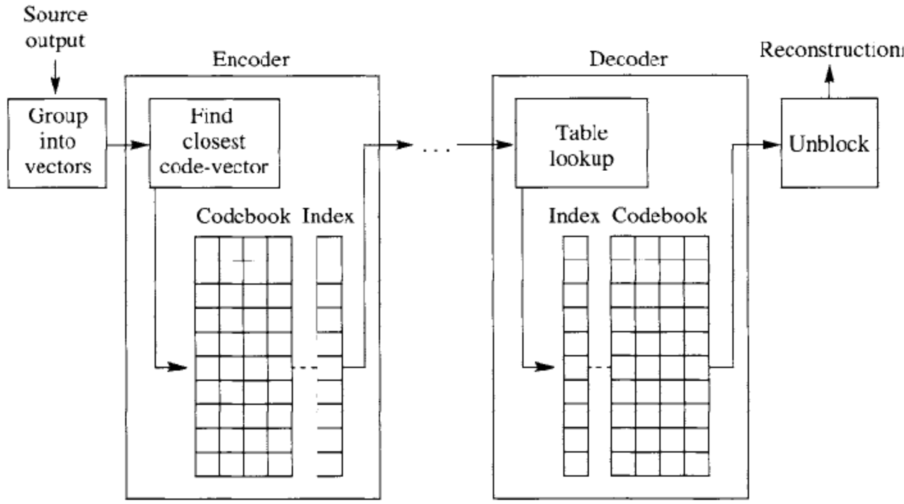

Fig. 1: Diagram of compression scheme

This paper is organized as follows: In the section 2, we present the feature extraction Mel Frequency Cepstral Coefficient. The section 3 describes the stages of vector quantizer based on the Self-Organizing Maps. Section 4 presents the proposed model for optimizing the architectures of Kohonen networks. The genetic algorithm to solve the obtained model is described in section 5. In the last section, some numerical results are reported.

2. Feature Extraction from speech (MFCC)

Voice Signal Identification consists of the process to convert a speech waveform into features that are useful for further processing. There are many algorithms and techniques are use. It depends on features capability to capture time frequency and energy into set of coefficients for cepstrum analysis [1].
Despite many differences between individuals, and the existence of many languages, speech follows general patterns, and on average has well defined characteristics such as those of volume, frequency distribution, pitch rate and syllabic rate [3]. These characteristics have adapted with regard to environment, hearing and voice production limitations, speech characteristics fit the speech generating abilities of the body, but the rapid changes in society over the past century have exceeded our ability to adapt.
The human voice is converted into digital signal form to produce digital data representing each level of signal at every discrete time step.
In this section, we describe the mechanism to convert the speech signal to some type of parametric representation.
A wide range of possibilities exist for parametrically representing the speech signal for the speech processing task, such as Linear Prediction Coding (LPC), Mel Frequency Cepstral Coefficient (MFCCs), and others.
Generally speaking, a conventional speech processing system can be organized in two blocks: the feature extraction and the modeling stage. In practice, the modeling stage is subdivided in acoustical and language modeling, both based on SOM as described in Fig. 2.

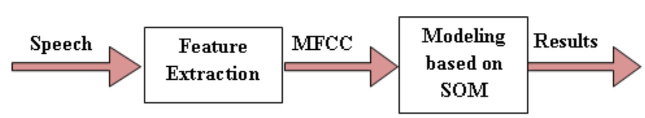

Fig. 2: Representation of speech processing by SOM

MFCCs are the most commonly used acoustic features in speech processing, and this feature has been used in this paper. MFCC's are based on the known variation of the human ear's critical bandwidths with frequency [3].

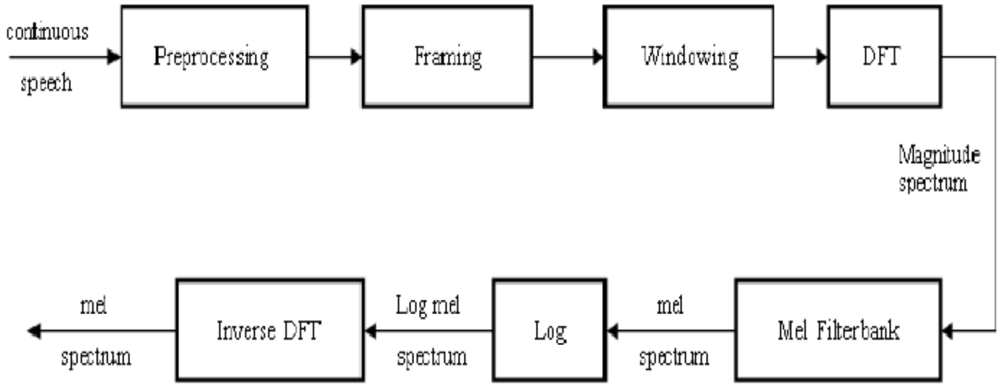

Fig. 3: Block diagram of the computation steps of MFCC

The Fig. 3 shows the seven steps to construct the parameters Mel Frequency Cepstral Coefficient [26]. Each step has its function and mathematical approaches as discussed briefly in the following:

- *Pre-processing*

This step processes the passing of signal through a filter which emphasizes higher frequencies. This process will increase the energy of signal at higher frequency.

- *Framing*

The framing permits to segment the speech samples obtained from analog to digital conversion into a small frame with the length within the range of 20 to 40 ms.

- *Hamming windowing*

Hamming window is used as window shape by considering the next block in feature extraction processing chain and integrates all the closest frequency lines.

- *Fast Fourier Transform*

This step consists to convert each frame of N samples from time domain into frequency domain using the Fourier.

- *Mel Filter Bank Processing*

The frequencies range in FFT spectrum is very wide and voice signal does not follow the linear scale. The bank of filters according to Mel scale as shown in Fig. 4 is then performed.

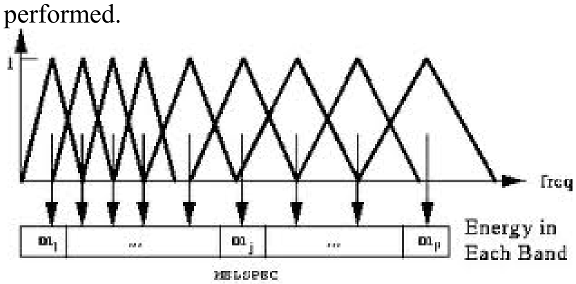

Fig. 4: Mel –scale Filter Bank

This figure shows a set of triangular filters that are used to compute a weighted sum of filter spectral components so that the output of process approximates to a Mel scale. Each filter's magnitude frequency response is triangular in shape and equal to unity at the centre frequency and decrease linearly to zero at centre frequency of two adjacent filters [9]. Then, each filter output is the sum of its filtered spectral components.

- *Discrete Cosine Transform*

This is the process to convert the log Mel spectrum into time domain using Discrete Cosine Transform (DCT). The results of the conversion are called Mel Frequency Cepstrum Coefficient. The set of coefficients are called acoustic vectors. Therefore, each input utterance is transformed into a sequence of acoustic vector.

- *Delta Energy and Delta Spectrum*

The voice signal and the frames changes, such as the slope of a formant at its transitions. Therefore, there is a need to add features related to the change in cepstral features over time. 13 delta or velocity features (12 cepstral features plus energy), and 39 features a double delta or acceleration feature are added.

3. Self-Organizing Maps and Vector Quantization

The Kohonen network is probably the closest of all artificial neural network architectures and learning schemes to the biological neuron network. The aim of Kohonen learning is to map similar signals to similar neuron positions.

The Kohonen network has one single layer of neurons arranged in a two-dimensional plane, let name this one the output layer (see Fig. 5). The additional input layer just distributes the inputs to output layer. The number of neurons on input layer is equal to the dimension of input vector. A defined topology means that each neuron has a

IJCSI International Journal of Computer Science Issues, Vol. 9, Issue 2, No 1, March 2012
ISSN (Online): 1694-0814
www.IJCSI.org

defined number of neurons as nearest neighbors, second-nearest neighbors, etc. The neighborhood of a neuron is usually arranged either in squares, which means that each neuron has either four nearest neighbors. Kohonen has proposed various alternatives for the automatic classification [20], and presented the Kohonen topological map; these models belong to the category of artificial neural networks to learning unsupervised without human intervention, the little information is necessary to respect the characteristics of input data. This model calculates simply a Euclidean distance between her input and weights [21].

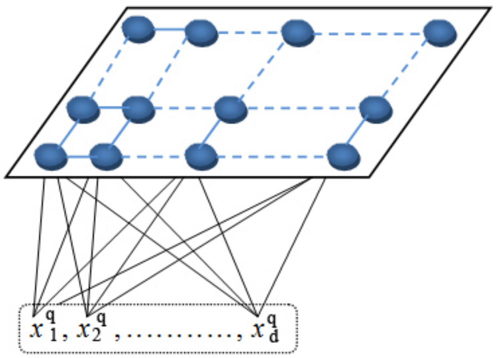

Fig. 5: Kohonen topological map.

To look at the training process more formally, let us consider the input data as consisting of n-dimensional vectors $X = \{x^1, x^2, \ldots, x^n\}$.

Meanwhile, each of N neurons has an associated reference vector $w^j = (w_1^j, w_2^j, \ldots, w_d^j)$.

During training, one x at a time is compared with all w^j to find the reference vector w^k that satisfies a minimum distance or maximum similarity criterion. Though a number of measures are possible, the Euclidean distance is by far the most common:

$$k = \arg min_{j=1}^{N} \|x - w^j\|$$

The best-matching unit (BMU) and neurons within its neighborhood are then activated and modified:

$$w^i(t+1) = w^i(t) + \beta_{i,k}(t)\|x - w^i\|$$

One of the main parameters influencing the training process is the neighborhood function ($\beta_{k,i}(t)$), which defines a distance-weighted model for adjusting neuron vectors. It is define by the following relations:

$$\beta_{k,i}(t) = \exp\left(\frac{-d_{k,i}}{2\sigma_i^2(t)}\right)$$

One can see that the neighborhood function is dependent on both the distance between the BMU and the respective neuron ($d_{k,i}$) and on the time step reached in the overall training process (t). In the Gaussian model, that neighborhood's size appears as kernel width (σ) and is not a fixed parameter. The neighborhood radius is used to set the kernel width with which training will start. One typically starts with a neighborhood spanning most of the SOM, in order to achieve a rough global ordering, but kernel width then decreases during later training cycles.

4. Proposed model to optimize the Kohonen architecture maps

The Kohonen method consists to project the high dimensional data in low dimensional space and conserve the notion of the neighborhood between the data.

Since the function minimized by the Kohonen method don't contain any term which control the size of the map. This later may contains same unnecessary neurons. Generally, if the size of the Kohonen map is chosen randomly the Kohonen learning algorithm gives tree classes of neurons as showing in Fig. 6, the first class that does represent any observation (empty class), the second class represents few information and last presents the important information for data.

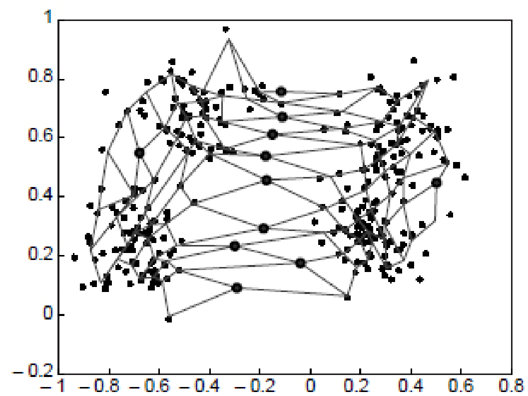

Fig. 6: Tree classes of neurons.

The neurons of the first and the second class have a negative effect because they make the learning process heavier. We model this problem of neural architecture in terms of a mixed-integer non linear problem with linear constraints, this modeling controls the size of the map, conserves the notion of neighborhood defined in the observations and a good initial weights.

In this section, we will describe the construction steps of our model. The first one consists in integrating the special term which control the size of the map. The second step gives the constraint which ensures the allocation of every data to only one neuron. For modeling, the problem of neural architecture optimization, we have needed to define some parameters as follows:

- n : Observation number of data base;
- p : Dimension of data base;
- N: Optimal number of artificial neurons in the Kohonen topological map;

- N_{min}: Minimal number of artificial neurons in the Kohonen topological map;
- N_{max}: Maximal number of artificial neurons in the Kohonen topological map;
- $X = \{x^1, x^2, \ldots, x^n\}$ Training base, where $x^k = (x_1^k, x_2^k, \ldots, x_d^k)$ for $k = 1, \ldots, n$.
- $U = (u_{i,j})$: The binary variables for $i = 1, \ldots, n$ and $j = 1, \ldots, N_{max}$, $u_{i,j} = 1$ if the i^{th} example is assigned to j^{th} neuron, else $u_{i,j} = 0$.

4.1 Objective function

The objective function of nonlinear programming model is a summation of product of a decision variable of each neuron in the Kohonen map, the matrix of neighbors and the distances between the observations and the neurons, this function is defined by:

$$E(U, W) = \sum_{i=1}^{n} \sum_{j=1}^{N_{max}} u_{i,j} K_j(\delta/T) \left\| x^i - w^j \right\|^2$$

Where $K_j(\delta/T) = exp(-\delta/T)$, δ represents the distance on the map between the referent neuron for the observation x^i and his neighbors, $W = \{w^1, w^2, \ldots, w^{N_{max}}\}$, $w^j = (w_1^j, w_2^j, \ldots, w_d^j)$ and $j = 1, \ldots, N_{max}$. If $u_{i,j} = 1$ then the i^{th} example x^i is assigned to the j^{th}neuron, and the corresponding error $\left\| x^i - w^j \right\|$ has been calculated on the objective function E.

4.2 Constraints

Each observation is assigned to a single neuron of the map, the constraint ensures that this assignment is:

$$\sum_{j=1}^{N_{max}} u_{i,j} = 1 \text{ for } i = 1, \ldots, n$$

4.3 Optimization Model

Since the size of the Kohonen topological map is randomly chosen, some neurons in the Kohonen topological map are a negative effect in the learning algorithm. To reduce this type of neurons, without losing of the learning quality, we propose to solve the following proposed model (P):

$$(P)\begin{cases} E(U, W) = \displaystyle\sum_{i=1}^{n} \sum_{j=1}^{N_{max}} u_{i,j} K_j(\delta/T) \left\| x^i - w^j \right\|^2 \\ \quad Subject\ to: \\ \displaystyle\sum_{j=1}^{N_{max}} u_{i,j} = 1 \qquad \text{for } i = 1, \ldots, n \\ u_{i,j} \in \{0,1\} \qquad j = 1, \ldots, N_{max}, i = 1, \ldots, n \\ \quad w^j \in \mathbb{R}^d \qquad j = 1, \ldots, N_{max} \end{cases}$$

Let (U^*, W^*) presents the solution of the optimization problem (P).

Many exact methods for solving mixed-integer non-linear programming (MINLPs) include innovative approaches and related techniques taken and extended from Mixed-integer programming (MIP). Outer Approximation (OA) methods [6] [10], Branch-and-Bound (B&B) [15] [29], Extended Cutting Plane methods [34], and Generalized Bender's Decomposition (GBD) [10] for solving MINLPs have been discussed in the literature since the early 1980's. These approaches generally rely on the successive solutions of closely related NLP problems. For example, B&B starts out forming a pure continuous NLP problem by dropping the integrality requirements of the discrete variables (often called the relaxed MINLP or RMINLP). Moreover, each node of the emerging B&B tree represents a solution of the RMINLP with adjusted bounds on the discrete variables.

The disadvantage of the exact solution methods mentioned above is that they become computationally intensive as the number of variables is increased throughout the procedure. Therefore, efficient heuristic methods are required to solve large-size instances accurately.

The heuristics methods for solving combinatorial optimization have now a long history, and there are virtually no well-known, hard optimization problems for which a meta-heuristic has not been applied. Often, meta-heuristics obtain the best known solutions for hard, large-size real problems, for which exact methods are too time consuming to be applied in practice.

5. Solving the obtained optimization model

Combinatorial problems arise in many application areas, such as scheduling, planning, design, finance, configuration, control, and the natural sciences. Their efficient solution is crucial to many managers, engineers, and scientists [23].

The combinatorial optimization is a problem in operational research, discrete mathematics and computer science. Its importance is justified partly by the great difficulty of optimization problems and also by many practical applications that can be formulated as a combinatorial optimization problem. Although combinatorial optimization problems are often easy to define, they are usually difficult to solve. Indeed, most of these problems belong to the class of NP-Complete problems and therefore do not have to date of effective algorithmic solution valid for all data [2].

In this paper we propose a method for solving our mixed integer non-linear programming problem (P) using genetic algorithm (GA) to determine a global optimal or local optimal solution. The penalty method was used to evaluate those infeasible chromosomes generated from genetic reproduction.

5.1 Solving the obtained optimization model using Genetic Algorithm (GA)

In this section, we will describe the genetic algorithms to solve the proposed model for Kohonen networks architecture optimization. To this end, we have coded individual by tree chromosomes; moreover, the fitness of each individual depends on the value of the objective function.
The Genetic Algorithm (GA) was introduced by J. HOLLAND to solve a large number of complex optimization problems [17]. Each solution represents an individual who is coded in one or several chromosomes. These chromosomes represent the problem's variables. First, an initial population composed by a fix number of individuals is generated, then, operators of reproduction are applied to a number of individuals selected switch their fitness. This procedure is repeated until the maximums number of iterations is attained. GA has been applied in a large number of optimization problems in several domains, telecommunication, routing, scheduling, and it proves it's efficiently to obtain a good solution [4]. We have formulated the problem as a non linear program with mixed variables.

Genetic algorithm

1. Choose the initial population of individuals
2. Evaluate the fitness of each individual in that population
3. Repeat on this generation
4. Select the best-fit individuals for reproduction
   a. Crossover and Mutation operations
   b. Evaluate the individual fitness of new individuals
   c. Replace least-fit population with new individuals

   Until termination (time limit, fitness achieved, etc.)

- Initial population

The first step in the functioning of a GA is, then, the generation of an initial population. Each member of this population encodes a possible solution to a problem.
The individual of the initial population are randomly generated, and $u_{i,j}$ take the value 0 or 1, and the weights matrix takes random values in space $[x_{min}, x_{max}]^p$ where $x_{min} = \min\{x_k^i\}$ and $x_{man} = \text{man}\{x_k^i\}$ where $k = 1, \dots, p$ and $i = 1, \dots, n$. Because all the observations are in the set $[x_{min}, x_{max}]^p$.

- Evaluating individuals

After creating the initial population, each individual is evaluated and assigned a fitness value according to the fitness function.
In this step, each individual is assigned a numerical value called fitness which corresponds to its performance; it depends essentially on the value of objective function in this individual. An individual who has a great fitness is the one who is the most adapted to the problem.
The fitness suggested in our work is the following function:

$$f(i) = \frac{1}{1 + E(i)}$$

Minimize the value of the objective function is equivalent to maximize the value of the fitness function.

- Selection

The application of the fitness criterion to choose which individuals from a population will go on to reproduce.
Where

$$P_i = \frac{f_i}{\sum_{j=1}^{n} f_j}$$

- Crossover

The crossover is a very important phase in the genetic algorithm, in this step, new individuals called children are created by individuals selected from the population called parents. Children are constructed as follows:
We fix a point of crossover, the parent are cut switch this point, the first part of parent 1 and the second of parent 2 go to child 1 and the rest go to child 2.
In the crossover that we adopted, we choose 2 different crossover points, the first for the matrix of weights and the second is for vector U.

- Mutation

The rule of mutation is to keep the diversity of solutions in order to avoid local optimums. It corresponds on changing the values of one (or several) value (s) of the individuals who are (s) chosen randomly.

6. Experiments results

5.1 Dataset Description

The experiments were performed using the Arabic digit corpus collected by the laboratory of automatic and signals, University of Badji-Mokhtar - Annaba, Algeria. A number of 88 individual (44 males and 44 females) Arabic native speakers were asked to utter all digits ten times.. Depending on this, the database consists of 8800 tokens (10 digits x 10 repetitions x 88 speakers). In this experiment, the data set is divided into two parts: a training set with 75% of the samples and test set with 25% of the samples. In this research, speaker-independent mode is considered [16].

Table 1: Digits

Arabic	*English*	*Symbol*
صفر	ZERO	'0'
واحد	ONE	'1'
اثنان	TWO	'2'
ثلاثة	THREE	'3'
أربعه	FOUR	'4'
خمسه	FIVE	'5'
ستة	SIX	'6'
سبعه	SEVEN	'7'
ثمانية	EIGHT	'8'
تسعه	NINE	'9'

Table 1 shows the Arabic digits, the first column present the digits in language Arabic, the second column present the digit in language English and the last column shows the symbol of each digit.

5.2 Experiments and discussion

To solve the optimization model (P), we propose a method using the genetic algorithm. The most theorical and logarithmical results are permit to determine the optimal number of artificial neurons in the Kohonen topological map, and the good initial matrix of weights. The proposed approach for optimization of the Kohonen topological map is tested to realize the training stage. The experiments results are presented in the Table 2, where N_{max} presents the initial number of neurons, N_{min} minimal number of neurons and N the optimal number of neurons in the map. N is determined by solving the model (P).

Table 2: Results of the Kohonen architecture optimization

Iterations	N_{min}	N_{max}	*N*
100	256	512	435
100	256	512	415
200	256	512	423
500	256	512	386
500	256	512	420

This theoretical approach has been tested on the problem of speech data compression, the performances of the compression algorithms are evaluated in terms of bit rate (bits per pixel) and Peak Signal-to-Noise Ratio (PSNR) is given by:

$$PSNR = 10\log_{10}\left(\frac{nX^2}{MSE}\right)$$

where n is the length of the reconstructed signal, X is the maximum absolute square value of the signal x and Mean Squared Error (*MSE*) is defined as follows:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(\hat{x}(i) - x(i))^2$$

where $\hat{x}$ is the original speech signal and x is the reconstructed speech signal.
Numerical results obtained by applying the improved method and the classical one (SOM) to dataset of Arabic digits are presented in the Table 3. This table list for males and females, the Arabic digits, the PSNR (N_{max}) calculated by a map of N_{max}=512 neurons and the PSNR(N) calculated by a new size the map of N=420 neurons which determined by the proposed approach.

Table 3: Results obtained for Arabic digit by both approaches

	Males		*Females*	
	PSNR(N_{max})	*PSNR(N)*	*PSNR(N_{max})*	*PSNR(N)*
Syfr 0	37.88	37.40	39.43	39.53
Wahid 1	30.15	29.98	33.97	33.72
Itnan 2	30.90	30.50	29.83	29.53
Talat'a 3	33.23	32.71	34.82	34.66
Arba'a 4	32.32	31.81	35.89	35.61
Khams'a 5	33.20	32.69	23.41	22.86
Sit'a 6	36.80	36.74	37.61	37.54
Seba'a 7	39.05	37.69	32.73	32.66
Thamni'a 8	35.03	34.27	33.29	33.24
Tisa'a 9	37.96	37.39	34.98	34.53

Recall that the proposed method contained in additional phase; this phase consists to solve the proposed model in order to remove the unnecessary neurons from the initial map. For example a map of 512 neurons we obtain a map of 420, the proposed approach can remove about 30 neurons from initial map to construct the optimal map for the Kohonen network.

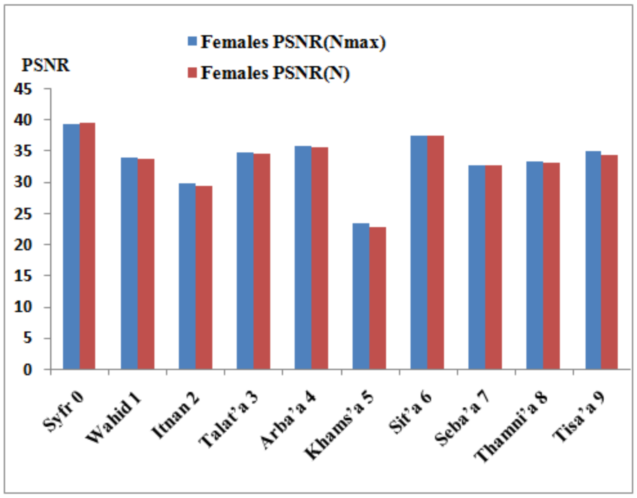

Fig. 7: Comparison between both approaches for the PSNR Females.

Fig. 7 and Fig. 8 show the PSNR comparison of Arabic Digits for the Females and the males between both algorithms, classical and proposed algorithm. We can see that the PSNR very close between both approaches. But proposed method can reduce the number of neurons and the training time.

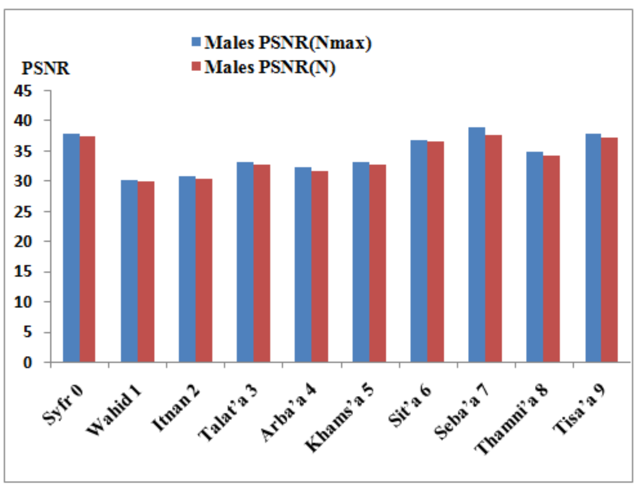

Fig. 8: Comparison between both approaches for the PSNR Males.

From a numerical point of view, our method permits to determine the optimal codebook from the optimal size of map. This optimal codebook gives same quality than the classical method, but with less compression time and reduces the memory space to store this codebook. Our method can also accelerate the phase of compression and decompression of a speech.

7. Conclusion

In this paper, we have proposed a new modeling for the Self Organizing Maps architecture optimization problems in terms of a mixed-integer nonlinear problem with linear constraints. The Genetic Algorithm (GA) is especially appropriate to obtain the optimal solution of the nonlinear problem. This method is tested to determine the optimal number of artificial neurons in the Kohonen map and the most favorable weights matrix.
This method has been compared to speech compression problem using a datasets of Arabic digit. The obtained results demonstrate the performance of the proposed method. The robustness of the proposed method to compress the speech is provided by the optimization of Kohonen architecture which determines the optimal codebook. In the next works, we will use exact approaches to resolve this problem and determine the optimal solution for the optimization of neural networks architectures. The proposed method can be applied to solve the pattern recognition problems, speech recognition problems and image compression problems.

References

[1] M. Ayache, M. Khalil and F. Tranquart. "Artificial Neural Network for Transfer Function Placental Development: DCT and DWT Approach". IJCSI International Journal of Computer Science Issues, Vol. 8, Issue 5, No 3, September 2011.
[2] J. F. Benders. "Partitioning procedures for solving mixed-variables programming problems". Numer. Math., 4:238, 1962.
[3] S. B. Davis and P. Mermelstein, "Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences", IEEE Trans. Acoust., Speech, Signal Process., vol. ASSP-28, no. 4, Aug. 1980, pp. 357–366.
[4] J. Dréo, A. Pétrowski, P. Siarry and E. Taillard. "Métaheuristiques pour l'optimisation difficile". Eyrolles, 2003.
[5] G. Dreyfus, J. Martinez, M. Samuelides, M. B. Gordon, F. Badran, S. Thiria, L. Hrault. "Réseaux de neurones Méthodologie et applications". Eyrolles 2ième édition.
[6] M. Duran, I. E. Grossmann. "An outer-approximation algorithm for a class of mixed-integer non linear programs". Mathematical Programming, 1986, pp. 307-339.
[7] M. Ettaouil. "A 0-1 Quadratic Knapsack Problem for Modelizing and Solving the Constraint Satisfaction Problems". In lecture Notes in Artificiel Intelligence N 1323, Springer Verlag, Berlin, 1997, pp. 61-72.
[8] M. Ettaouil, Y. Ghanou, K. El Moutaouakil, M. Lazaar, "Image Medical Compression by A new Architecture Optimization Model for the Kohonen Networks". International Journal of Computer Theory and Engineering vol. 3, no. 2, 2011, pp. 204-210.

[9] M. Ettaouil, Y. Ghanou. "Neural architectures optimization and Genetic algorithms". Wseas Transaction On Computer Issue 3, 2009, pp. 526-537.
[10] R. Fletcher and S. Leyffer. "Solving Mixed Integer Programs by Outer Approximation", Math. Program. 66, 1994, 327–349.
[11] A.M. Geoffrion. "A Generalized Benders Decomposition", J. Optim. Theory Appl., 10 (4), 1972, 237–260.
[12] S. Ghorbel, M. Ben Jemaa and M. Chtourou. "Object-based Video compression using neural networks". IJCSI International Journal of Computer Science Issues, Vol. 8, Issue 4, No 1, July 2011.
[13] K. Godfrey, Y. Attikiouzel. "Self-organized color image quantization for color image data compression". Proc ICNN 3, 1993, pp. 1622–1626.
[14] GrayRM. "Vector quantization". IEEE ASSP Magazine, 1984, pp 4–29.
[15] O.K. Gupta and A. Ravindran. "Branch and Bound Experiments in Convex Nonlinear Integer Programming", Manage Sci., 31 (12) , 1985, pp. 1533–1546.
[16] N. Hammami, M. Bedda , "Improved Tree model for Arabic Speech Recognition", Proc.IEEE, ICCSIT10, 2010.
[17] J. Holland. "Adaptation in natural and artificial systems". Ann Arbor, MI: University of Michigan Press, 1992.
[18] K.S.Kavitha, K.V.Ramakrishnan, Manoj Kumar Singh. "Modeling and design of evolutionary neural network for heart disease detection". IJCSI International Journal of Computer Science Issues, Vol. 7, Issue 5, September 2010.
[19] O.Khalifa, S.Khan, M.R.Islam, M.Faizal and D.Dol. "Text Independent Automatic Speaker Recognition". 3rd International Conference on Electrical & Computer Engineering, Dhaka, Bangladesh, 2004, pp.561-564.
[20] T. Kohonen. "Self Organizing Maps". Springer, 3th edition, 2001.
[21] T. Kohonen. "The Self Organizing Maps". Proceedings of IEEE, 78, No. 9, 1990, pp. 1464-1480.
[22] T. Kohonen, S. Kaski, K. Lagus, J. Salojr , J. Honkela, V. Paatero, A. Saarela. "Self organization of a massive document collection". IEEE transaction on neural networks, 11, No. 3, 2000.
[23] G. R. Kocis, I. E. Grossmann. "Global Optimization of Non convex Mixed-Integer Nonlinear Programming (MINLP) Problems in Process Synthesis". Ind. Eng. Chem. Res., 27, 1988, pp. 1407-1421.
[24] Linde Y, Buzo A, Gray RM. "An algorithm for vector quantizer design". IEEE Trans Commun 1, 1980, pp. 84–95.
[25] R. P. Lippmann, "Review of neural networks for speech recognition", Neural Computing, vol. 1, 1989, pp. 1- 38.
[26] L. Muda, M. Begam, and I. Elamvazuthi, "Voice Recognition Algorithms using Mel Frequency Cepstral Coefficient (MFCC) and Dynamic Time Warping (DTW) Techniques". Journal Of Computing, Volume 2, Issue 3, March 2010.
[27] M. Oral, O. Kettani. "A linearization procedure for quadratic and cubic mixed-integer problems". University Laval, Quebec, Canada, 40, No. 1, 1992.
[28] S. Papadimitriou, S. Mavroudi, L. Vladutu, G. Pavlides, A. Bezerianos. "The supervised network self-organizing map for classification of large data sets". Applied intelligence 16, Kluwer Academic publishers. Manufactured in the netherlands, 16, 2003. pp. 185-203
[29] I. Quesada and I.E. Grossmann. "An LP/NLP Based Branch and Bound Algorithm for Convex MINLP Optimization Problems", Computers Chem. Eng., 16 (10/11), 1992, pp. 937–947.
[30] D. Salomon, "Data compression", Springer, 2004.
[31] K. Sayood, "Introduction to Data Compression", Elsever, 3th Edition. 2006.
[32] D. Vigliano, R. Parisi, A. Uncini, "Video Compression by Neural Networks", Intelligent Multimedia Processing with Soft Computing, Vol. 168, 2005, pp. 205-234.
[33] D. Wang. "Fast Constructive-Coverting Algorithm for neural networks and its implement in classification". Applied Soft Computing, 8, 2008, pp. 166-173.
[34] T.Westerlund and F. Petersson. "A Cutting Plane Method for Solving Convex MINLP Problems", Computers Chem. Eng., 19, 1995, pp. 131–136.