

# ON THE OPTOELECTRONIC IMPLEMENTATION OF NEURAL NETWORKS. SOURCES OF ERRORS AND THEIR INFLUENCE ON THE INFLUENCE OF RANDOM DEVIATIONS OF THE WEIGHTS ON THE PERFORMANCES

Ioan Ileană, Joldeş Remus

**Abstract.** One of the most promising implementation of neural networks is optoelectronic implementation. Optical interconnections are useful for neural networks as far as one can take advantage of the special potential of 3D connection through free space. This paper analyses the consequences caused by random deviations of the neurons interconnection weights from the accurately computed values. For the neural network we used, the connections are considered to be implemented optically, by computer generated holograms (CGH) and we outlined the main causes of weight deviations. Also we study the influence of these deviations on the network's dynamics, accuracy of prototype recall, spurious states etc. The theoretical results are sustained by simulations of a concrete autoassociative neural network.

**Keywords:** Autoassociative memory, weights matrix, random deviations, pattern recognition.

## 1. INTRODUCTION

The artificial neural networks have the specific feature of "storing" the knowledge in the synaptic weights of the processing elements (artificial neurons). There is a great number of algorithms allowing the design of neural networks (e. g.<sup>2</sup>) and the computing of weight values.

Depending on the network type and the purpose had in sight, the weights can be established through an iterative learning process (supervised or unsupervised) or they can be computed from the apriori knowledge on the input and output network space. In software implementations, the weights computed according to a certain algorithm will be used with precise, accurate values, resulting from calculus (although there might also appear problems caused by data precision - the number of bytes assigned to one value). In hardware implementations of neural networks, the weights values will be materialized in a technological process during which various errors may occur, so that the resulting network will use more or less deviated weights.

It is expected that these deviations of the weights from the accurate values influence the network's behaviour during the working phase. This paper therefore intends an analysis of the consequences of synaptic weights deviations from the accurate values in the particular case of an associative network, whose neurons are optically interconnected through CGH.

Section 2 briefly presents the network model we used, as well as some results concerning the behaviour of this network type. Section 3 reviews the main causes of errors that may occur when using CGH to implement the interconnections of neurons. Section 4 realises a detailed study on the consequences of the random weights deviations on the behaviour of the network and section 5 shows the results of some simulations realised by the authors.

## 2. THE ARTIFICIAL NEURAL NETWORK MODEL

In this section we'll briefly introduce the main theoretical elements concerning associative memories and recurrent neural networks<sup>4</sup>, elements that will be used in section 5 in order to build an associative memory for patterns recognition.

We'll call *pattern* a multidimensional vector with real components. An associative memory is a system that accomplishes the association of p pattern pairs  $\xi^\mu \in R^n$ ,  $\zeta^\mu \in R^n$ , ( $\mu=1, 2, \dots, p$ ) so that when the system is given a new vector  $x \in R^n$  such as

$$d(\mathbf{x}, \xi^i) = \min_j d(\mathbf{x}, \xi^j) \quad (1)$$

the system responds with  $\zeta^i$ ; in (1)  $d(\mathbf{a}, \mathbf{b})$  is the distance between patterns  $\mathbf{a}$  and  $\mathbf{b}$ .

The pairs  $(\xi^i, \zeta^i)$ , ( $i=1, 2, \dots, p$ ) are called prototypes and the association accomplished by the memory can be defined as a transformation  $\Phi: R^n \times R^m$  so that  $\zeta^i = \Phi(\xi^i)$ .

The space  $\Omega \subset R^n$  of input vectors  $\mathbf{x}$  is named configuration space and the vectors  $\xi^i$ , ( $i=1, 2, \dots, p$ ) are called attractors or stable points. Around each attractor, there is a basin of attraction  $B_i$  such that  $\forall x \in B_i$ , the dynamics of the network will lead to the stabilization of  $(\xi^i, \zeta^i)$  pair. For autoassociative memories  $\xi^i = \zeta^i$  ( $i=1, 2, \dots, p$ ) and if some vector  $\mathbf{x}$  is nearest  $\xi^i$ , then  $\Phi(\mathbf{x}) = \xi^i$ .

In section 4 we will use for graphic pattern recognition a neural network whose model<sup>4</sup> is presented below. Let's consider the single-layer neural network built from totally connected neurons, whose states are given by  $x_i \in \{-1, 1\}$ ,  $i=1, 2, \dots, n$ , (fig.1).

We denote:  $\mathbf{W}=[w_{ij}, 1 \leq i, j \leq n]$  the weights matrix,  $\theta=[\theta_1, \dots, \theta_n]^T \in R^n$  the thresholds vector,  $\mathbf{x}(t)=[x_1(t), \dots, x_n(t)]^T \in \{-1, 1\}^n$  the network state vector.

For the memory described in this paper, the weight matrix  $\mathbf{W}$  will be built as follows: given a set of n-dimensional prototype vectors  $\mathbf{X}=[\xi^1, \xi^2, \dots, \xi^p]$ , we establish the synaptic matrix  $\mathbf{W}$  and the threshold vector  $\theta$ , so that the prototype vectors become stable points for the associative memory, that is:

$$\xi^i = \text{sgn}(\mathbf{W}\xi^i - \theta) \quad i = 1, 2, \dots, p \quad (2)$$

where the sgn function is applied to each component of the argument.

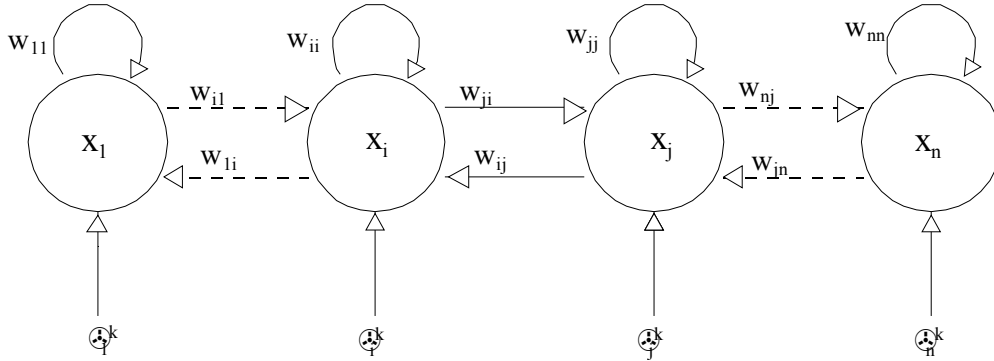


Fig. 1. Single layer recurrent neural network

We use the Lyapunov method to study the stability of the network, defining an energy function and showing that this function is decreasing for any network trajectory in the state space. In our case, we'll consider the following energy function:

$$E(\mathbf{x}) = -\frac{1}{2} \mathbf{x}^T \mathbf{W} \mathbf{x} - \mathbf{x}^T \boldsymbol{\theta} \quad (3)$$

where  $\mathbf{x}$  is the network state vector at time  $t$ .

From the evolution in states space point of view, a network can reach a stable point or can execute a limit cycle. At its turn, a stable point can be a global or a local energy minimum. Using (3), the following results concerning the network's dynamics are demonstrated in<sup>4</sup>:

- a) The asynchronous updating mode:
  - X If the synaptic matrix is symmetric and has non-negative diagonal elements, the recurrent network in asynchronous operation has only stable states (there are no cycles). As a consequence, the different variants of Hebb's law (weighted, with or without null diagonal) and the projection rule guarantee the absence of cycles for asynchronous mode.
- b) The synchronous updating mode:
  - X If the synaptic matrix is defined non-negative on  $\{-1,0,1\}^n$ , the recurrent network in synchronous functioning has no cycles. As a consequence, Hebb's rule (with or without weighting) and projection rule assure the absence of

cycles in synchronous mode. If the diagonal is null, two-length cycles may appear.

### 3. ERROR SOURCES FOR OPTICAL INTERCONNECTION WITH CGH

Optical interconnections are useful for neural networks as far as one can take advantage of the special potential of 3D connection through free space. This involves the organizing of the neurons layer in 2D configurations (planar), where the optical interconnections realise the desired links between the two planes. A certain connection also materialises the synaptic weight corresponding to neuron  $j$  from input plane and neuron  $i$  from output plane (fig. 2).

The interconnection network accomplishes the following function:

$$\beta(k,l) = \sum_{ij} \alpha(i,j) \cdot T_{ijkl} \quad (4)$$

In order to connect the two neuron planes one may use computer generated holograms which, by light waves diffraction, assure the desired connections. Due to the fact that generally the connections differ from neuron to neuron, the interconnection system will be a spatial variant system, each point from the input plane being connected differently to the output plane. This spatial variance can be realised in several manners.

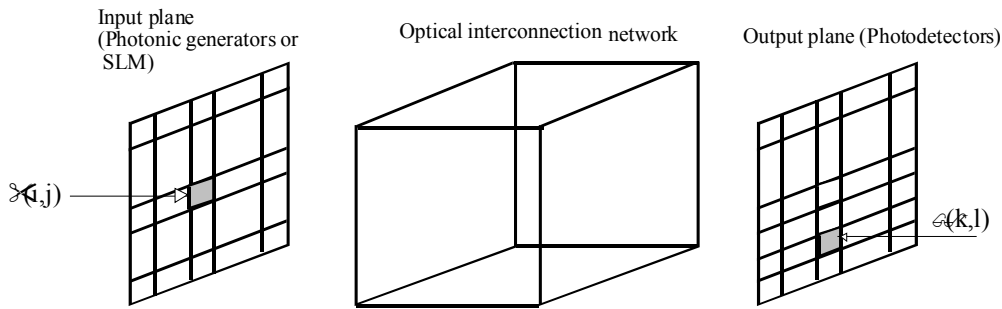


Fig. 2 .Optical interconnection of two neural planes.

Implementing optical interconnection with CGH implies the following steps:

- synthesis<sup>8</sup> of CGH for the synaptic weights matrix  $\mathbf{W}$  (fig. 3)

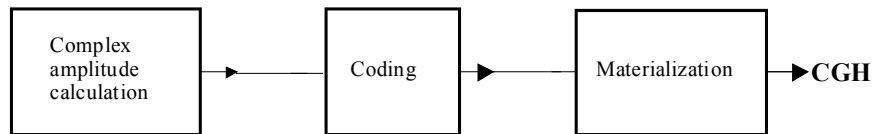


Fig. 3. Main steps in CGH realisationbuilding and aligning the optical setup.

In the following we will briefly review the errors that may occur in the optical implementation of interconnections, leading to deviations of the real weight values from the accurate ones.

**A.** During the CGH computing step, the errors are due to the hologram synthesis method, as well as to the quantization of the obtained values<sup>7, 8,10,11</sup>. When using Fourier amplitude holograms, a frequent difficulty is connected to the dynamic range of Fourier coefficients to be materialized in hologram. The computed range would lead to a smaller number of large aperture cells, most of the apertures being small, which significantly diminishes the hologram efficiency on diffraction. Consequently, a dynamic range compression is required, with direct influence on the accuracy of the resulting synaptic weights.

During the CGH materialization step, the errors may occur from the unsatisfactory resolution of the devices involved or from the photolithographic process.

**B.** During the building of optical setup (the hardware implementation of the neural network), the following error sources may appear:

- Un-uniform optical emission of the photoemitters from the output plane of the neural layer (the input plane of the optical interconnection system)
- Un-uniform sensitivity of the photodetectors in the input plane of the neural layer (the output plane of the optical interconnection system)
- Axial misalignment (translation or rotation) of the three planes: the input plane, the hologram plane and the output plane.
- Errors concerning the positioning of individual photoemitters or photodetectors in their corresponding plane.
- Errors due to the variation of the geometrical dimensions of the setup, as a result of temperature variations.
- Errors caused by interaction of the adjacent optical paths (the light flow deviated by the hologram to a certain photodetector also reaches the nearby photodetectors).

All the errors above can be minimized, but it is obvious they cannot be completely removed. Consequently, we find very useful an analysis of the influences these errors might have on the global behaviour of the network.

#### 4. INFLUENCE OF RANDOM WEIGHT DEVIATIONS

In the following, we will present a general qualitative study on the effects of the weight deviations from the accurate values. We'll denote by  $\mathbf{W}$  the accurate (theoretical) weight matrix and by  $\mathbf{W}^*$  the real weight matrix. We may then consider:

$$\mathbf{W}^* = \mathbf{W} + \delta\mathbf{W} \tag{5}$$

where  $\delta\mathbf{W}$  is a matrix of deviations from the computed values.

As we have shown in section 2, an energy function (3) is defined for the neural

network. In order to simplify the computing we will consider null thresholds for all the neurons. The network's dynamics is associated to this energy function and one can easily see that the weight values deviations will have important consequences on the network's functioning.

Thus, the random deviations of synaptic weights values can cancel the symmetry of  $\mathbf{W}$  matrix and the property of diagonal elements (e. g. they may become negative), making the results in section 2 (concerning stability) no longer valid. A non-symmetric or negative diagonal weight matrix may lead to the occurrence of cycles in the network's evolution. Therefore, the network won't reach a stable state, consequently it won't work well as associative memory.

The weight matrix synthesis starts from a set of prototype vectors we wish to be stored. From energy perspective, these prototypes are global minimum of function  $E$  (of value  $E_0$ ). If instead of the accurate  $\mathbf{W}$  matrix the network operates using the  $\mathbf{W}^*$  matrix, the energy value for some prototype becomes:

$$E(\xi) = -\frac{1}{2}\xi^T \mathbf{W} \xi - \frac{1}{2}\xi^T \delta \mathbf{W} \xi = E_0 + \delta E \quad (6)$$

One may observe that the weights deviation from accurate values will modify the "energy landscape". The prototypes minimum may change, as well as spurious states minimums may occur.

Up to this point we didn't make any assumption on the range or statistical distribution of deviations. Now we will perform, following<sup>3</sup>, a more detailed analysis, to see the influence of deviations on the maximum number of prototypes the memory can store. For this, starting from the dynamic equation with  $\theta_i = 0$ , we'll study the stability of component  $\xi_i^\nu$  ( $i=1,2,\dots,n$ ) of the prototype  $\xi^\nu$ :

$$\xi_i^\nu = \text{sgn}\left(\sum_{j=1}^n w_{ij}^* \xi_j^\nu\right) = \text{sgn}\left(\sum_{j=1}^n w_{ij} \xi_j^\nu + \sum_{j=1}^n \delta w_{ij} \xi_j^\nu\right) \quad (7)$$

We'll call net input (or synaptic potential):

$$h_i^\nu = \sum_{j=1}^n w_{ij}^* \xi_j^\nu = \sum_{j=1}^n w_{ij} \xi_j^\nu + \sum_{j=1}^n \delta w_{ij} \xi_j^\nu \quad (8)$$

Assuming that the weights have been computed using Hebb's rule, it is easy to outline the contribution of  $\xi_i^\nu$  to net input:

$$h_i^\nu = \xi_i^\nu + \frac{1}{n} \sum_{j=1}^n \sum_{\mu \neq \nu} \xi_i^\mu \xi_j^\mu \xi_j^\nu + \sum_{j=1}^n \delta w_{ij} \xi_j^\nu \quad (9)$$

where  $\mu := 1, 2, \dots, p$  indexes the  $p$  prototypes.

We denote by  $Z_i$  the additive term from (14) and we make the observation that it doesn't have the same sign with  $\xi_i^\nu$ , it may make unstable this component of the prototype  $\nu$ . To examine the influence of random deviations on maximum number of

prototypes that may be stored, we'll consider:

$$C_i^v = -\xi_i^v \cdot Z_i = -\xi_i^v \left( \frac{1}{n} \sum_{j=1}^n \sum_{\mu \neq v} \xi_i^\mu \xi_j^\mu \xi_j^v + \sum_{j=1}^n \delta w_{ij} \xi_j^v \right) \quad (10)$$

The probability for  $\xi_i^v$  to be unstable is (remember that  $\xi_i$  may take only -1 or +1 values):

$$p_{unst} = Prob(C_i^v > 1) \quad (11)$$

We shall try to find the distribution function and the statistical parameters for the random variable  $C_i^v$ .

Let

$$A_i = -\xi_i^v \frac{1}{n} \sum_{j=1}^n \sum_{\mu \neq v} \xi_i^\mu \xi_j^\mu \xi_j^v \quad (12)$$

and

$$B_i = -\xi_i^v \sum_{j=1}^n \delta w_{ij} \xi_j^v \quad (13)$$

In the reference [3] it is shown that the variable  $A_i$  obeys, with a good approximation, a gaussian distribution with mean 0 and variance  $p/n$ . We'll consider that the deviations  $\delta W_{ij}$  also obey a gaussian distribution with mean 0 and variance  $\sigma_\delta^2$ . In this case, the variable  $B_i$  will obey, with a good approximation, a gaussian distribution with mean 0 and variance  $\sigma_\delta^2$ . Consequently, we may assume that variable  $C_i$  obeys a normal distribution with mean 0 and variance:

$$\sigma^2 = \frac{p}{n} + \sigma_\delta^2 \quad (14)$$

We denote

$$p' = p + n \cdot \sigma_\delta^2 \quad (15)$$

and using the results from reference [3] we find that the correct functioning of the network implies

$$\frac{p'}{n} \leq 0.138 \quad (16)$$

or

$$p \leq (0.138 - \sigma_\delta^2) \cdot n \quad (17)$$

One can thus see that **random deviations of the weights can seriously affect the maximum number of prototypes** that may be stored in the network, as well.

5. EXPERIMENTAL RESULTS AND CONCLUSIONS

We've performed a large number of simulations in order to verify the theoretical considerations from the precedent section. We've studied an autoassociative neural network, synthesized starting from a set of 10 prototypes representing the cursive script numerals 0..9 (fig. 4). The simulations have had in view the influence that random weight deviations have on the following network features:

- Noise tolerance. The network had to recognize noise affected prototypes, using the accurate  $\mathbf{W}$  matrix, and also the deviated weights matrix  $\mathbf{W}^*$ .
- The energy minimums corresponding to prototypes. For the 10 prototypes we have computed the energy minimums, in the accurate weights phase, as well as using the deviated weights.
- The average number of iterations before stabilizing into an attractor.

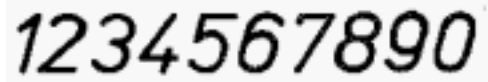


Fig.4. The prototypes used in simulations

The results we've obtained are illustrated below:

- When the associative memory operates with the accurate weights, it shows a very good noise immunity, recognizing prototypes affected by up to 40% noise (fig. 5).



Fig. 5. The recognition of noisy prototypes by the accurate network

If the weight matrix used is  $\mathbf{W}^*$ , the noise immunity decreases, so that the network is no longer able to recognize prototypes with the same noise contamination (fig.6).

In order to study the above mentioned objectives, we've performed several simulations affecting the weight matrix  $\mathbf{W}$  with random deviations having various distribution laws and parameters.

A first set of simulations pursued the altering of the accurate matrix with random deviations having a gaussian distribution, with different mean and variance values. For each case, we've performed 100 recognitions of every prototype, disturbed by 40% noise. The results are shown in fig. 7.

A second set of simulations included the usage of a matrix in which every weight has been randomly affected by an uniformly distributed error between 5% and 10% of



its value. In this case also we've performed 100 recognitions for each noise affected prototype. The results are shown in table 1.

II. For every deviated matrix we used, we've computed the prototype corresponding energy values, watching the energy minimum's variations. The simulations outlined, for the above cases, a slight disturbance of the energy minimums (about 3% of the values corresponding to the accurate weights). Also, we've noticed that the average number of iterations slightly varies from the value of 3, no matter which the weights deviations are.



Fig. 6. Failed recognition of prototypes by the deviated weights network

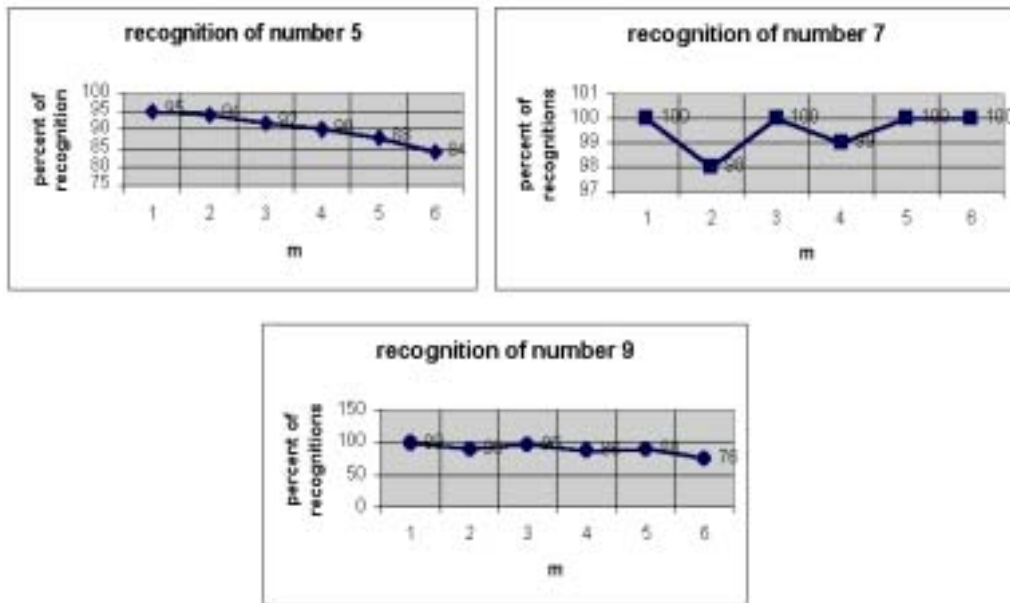


Fig. 7. Recognition of some noisy prototypes by the network with weights disturbed by Gaussian deviations with variance  $\Phi_w^2 - m^2 \Phi_w^2$

Table 1.

ACTA UNIVERSITATIS APULENSIS

Prototype	0	1	2	3	4	5	6	7	8	9
Percent of correct recognitions	94	100	94	81	95	86	98	100	98	91

**CONCLUSIONS**

The simulations have shown diminished performances of the autoassociative memory as a result of the random weights deviations from the correctly computed values. For the statistical parameters used above the performances' degradation is relatively small, which proves certain insensitivity to those deviations. Therefore, the memory appears quite robust, not only in what the noise contained by the patterns to be recognized is concerned, but also in relation with the random weights deviations.

It is possible for this insensitivity to occur also due to the fact that the number of stored prototypes is much lower than the memory's capacity. The simulations performed on another associative memory, designed to recognize cursive script letters, confirm this last assumption. In this case the input vector's size is still of 1024 components, but the amount of stored prototypes becomes 27.

This aspect is to be elucidated in further studies. It would also be useful to determine quantitatively the contribution of the errors described in section 4 to the deviation of actual weights from accurate ones, in order to state realistic requests for the hardware implementation of the autoassociative memory.

**REFERENCES**

1. Cox J. A., Werner T., Lee J., Nelson S., Fritz B. and Bergshom J.: "Diffraction efficiency of binary optical elements. Proc. SPIE vol. 1211, pp.116-124 (1990).
2. Dumitra[] Adriana: *Proiectarea re $\leftrightarrow$ elelor neuronale artificiale*, Casa editorial| Odeon, Bucure[]ti, 1997.
3. Hertz John, Krogh Anders, Palmer G. Richard: *Introduction to the theory of neural computation*, Addison-Wesley Publishing Company, 1991.
4. Kamp Yves, Hasler Martin: *Reseaux de neurones r $\leftrightarrow$ cursifs pour m $\leftrightarrow$ moires associatives*, Presses polytechniques et universitairea romandes, Lausanne, 1990.
5. Keller E. Paul and Gmitro F. Arthur: "Computer-generated holograms for optical neural networks: on-axis versus off-axis geometry", *Applied Optics* 32, pp.1304-1310 (10 March 1993).
6. Keller E. Paul and Gmitro F. Arthur: "Design and analysis of fixed planar holographic interconnects for optical neural networks". *Applied Optics* 31, pp.5517-5526, 10 September 1992.
7. Lee Hon Wai: "Sampled Fourier Transform Hologram Generated by Computer", *Applied Optics vol 9 (3) pp.639-643* (march 1970).

8. Lohmann A. W. and Paris D. P.: "Binary Fraunhofer Holograms, Generated by Computer, *Applied Optics*, Vol 6 (10), pp.1739-1745 (October 1967).
9. Riehl J., Appel J. Thiriot A., Dorey J.: "Hybrid electronic/non coherent optical processor for large scale phased arrays". SPIE vol. 963 Optical Computing 88, pp.337-345.
10. Seldowitz A. Michael, Allebach Jan P., and Sweeney W. Donald: "Synthesis of digital holograms by direct binary search", *Applied Optics*, vol 26, No. 14/ 15 July 1987, pp.2788-2798.
11. Wolf E.: *Progress in Optics XXVIII*, Elsevier Science Publishers B. V. 1990.

**Authors**

Remus JOLDEȘ, Ioan ILEANĂ, Universitatea "1 Decembrie 1918" Alba Iulia, email [rioJdes@uab.ro](mailto:rioJdes@uab.ro) , [iileana@uab.ro](mailto:iileana@uab.ro)