Multi-particle Networks for Associative Memory

Dmitri N. Golub[1] and Alexander N. Gorban[2]

[1]**Krasnoyarsk State University, Krasnoyarsk, Russia**
[2]**Computing Center, Akademgorodok, Krasnoyarsk, 660036, Russia**
e-mail : amse@cckr.krasnoyarsk.su

New light is thrown on Associative Memory Networks. Multi-particle or high-rank tensor approach, which is a generalization of the Hopfield model, is considered for the Pattern Recognition. The approach allows to enhance significantly informational capacity of Associative Memory Networks. A new efficient method is offered for calculating high order or multi-particle nets. The method is confirmed by two experiments, the results of which are presented and discussed in the paper.

Key-words : Associative Memory, Neural Network, k-particle Net.

1. Introduction

The fundamental problem in the theory of Associative Memory is the enhancement of informational capacity. The desired goal is to obtain networks which would reliably memorize a great number of images. Our basic concept is to take into consideration multi-particle or high-rank tensors. It can be given the following interpretation. The Hopfield net [1] in our terminology corresponds to one-particle nets. The recognition procedure of each coordinate, in this case, is based on the coincidence of single points. Two-particle nets are based on the coincidence of pairs of points, and so on. In this light it becomes evident that multi-particle nets are more reliable in the recognition of patterns. It must be emphasized, however, that it has nothing to do with real particles, but is only a convenient interpretation. In the following sections it will be given rigorous mathematical description.

In other words, by using multi-particle vectors or high-rank tensors we increase the order of non-linearity of nets which we will denote the order of non-linearity by the letter k. We will denote images and patterns as bracketed (cket- $\left|\;\right\rangle$ and bra- $\left\langle\;\right|$) vectors. By a k-particle vector $\left|F^k\right\rangle$ of a vector $\left|X\right\rangle$ we will understand the tensor product of $\left|F^k\right\rangle = \underbrace{\left|X\right\rangle \otimes \ldots \otimes \left|X\right\rangle}_{k\ times}$.

A typical difficulty arising in the process of pattern recognition is the formation of “chimeras”or “spurious” images, which may be a superposition of several patterns. Our assumption is that the increase of the order of non-linearity of nets will enhance informational capacity and diminish the formation of “spurious” images. This issue will be discussed in section 4.

2. Construction of Multi-particle Networks and Recognition Procedure

Suppose we are given N patterns $\left|X_p\right\rangle$ to be memorized by the network. We will assume that all the components $\left\langle i \middle| X_p\right\rangle$ of the patterns will be confined to 1 and -1. Index p stands here for the p-*th* pattern. We would like to construct k-particle nets A^k with the property that

$$A^k\left|F_p^k\right\rangle = \left|X_p\right\rangle \tag{2.1}$$

To construct such k-particle nets we must take the following steps. First, we transfer the given patterns into the corresponding k-particle vectors $\left|F_p^k\right\rangle$. Second, we construct for each k the corresponding orthogonal set of k-particle vectors $\left|V_p^k\right\rangle$ such that $\left\langle V_p^k \middle| F_q^k\right\rangle = \delta_{pq}$, where δ_{pq} is the Kronecker symbol. The algorithm for constructing the k-particle vectors $\left|V_p^k\right\rangle$ is described in the Appendix A.

Once we have such k-particle vectors $\left|V_p^k\right\rangle$, we can write the k-particle net as the sum

$$A^k = \sum_{p=1}^{N}\left|X_p\right\rangle\left\langle V_p^k\right| \tag{2.2}$$

Evidently that (2.1) is satisfied: $A^k\left|F_q^k\right\rangle = \sum_{p=1}^{N}\left|X_p\right\rangle\delta_{pq} = \left|X_q\right\rangle$.

After the k-particle nets A^k are found, we may organize the recognition procedure in the following way. We take an "unknown" vector $|X\rangle$ and transfer it into the k-particle vector $|F^k\rangle$. Then we input it into the net A^k. This will produce another vector $A^k|F^k\rangle = |X'\rangle$. In general, the components of the obtained vector will not be integers. Therefore we will make binarization, i.e. we will put all coordinates of the vector $\langle i|X'\rangle$ to the $sign(\langle i|X'\rangle)$, thus returning the vector to the original vector space. Then we repeat it and obtain some other vector $|X''\rangle$. The process is stopped when the vector arrives at some final equilibrium point.

For simplicity we take into consideration only two outcomes. We count it a "success" if an "unknown" vector arrives to one of its nearest patterns. Otherwise , it is a "failure". There are two methods of how to organize such a process.

The first one is to actually construct all the k-particle $|F_p^k\rangle$ vectors, make their orthogonal sets of k-particle vectors $|V_p^k\rangle$ and construct the corresponding k-particle nets A^k in the manner just outlined above.

The other method allows to avoid actual construction of k-particle vectors. The idea is to directly calculate the coefficients C_p^k which occur during the iterative procedure:

$$A^k|F^k\rangle = \sum_{p=1}^{N} |X_p\rangle\langle V_p^k|F^k\rangle = \sum_{p=1}^{N} C_p^k |X_p\rangle \tag{2.3}$$

It can be easily shown (see Appendix A and B) that the coefficients C_p^k are given by the following formula

$$C_p^k = \frac{1}{\det\left(\langle X_i|X_j\rangle^k\right)} \sum_{q=1}^{N} M_{qp}^k \langle X_q|X\rangle^k \tag{2.4}$$

where the $N \times N$ matrix M_{ij}^k is composed of cofactors of the matrix $\left(\langle X_i|X_j\rangle^k\right)$. Taking it into account, we can write the expression (2.3) for the k-particle nets in the form

$$A_k|F^k\rangle = \frac{1}{\det\left(\langle X_i|X_j\rangle^k\right)} \sum_{p=1}^{N}\sum_{q=1}^{N} |X_p\rangle M_{qp}^k \langle X_q|X\rangle^k \tag{2.5}$$

The above formula allows to quickly calculate output images, for any values of k, without the actual construction of the k-particle vectors.

3. Experimental Results

We have carried out two tests.

In the tests we restricted ourselves only to the case of k=2,3,4,5, which is quite enough to see the general tendency. The size of the vector space was taken n=100 in the both tests.

As mentioned above, we used only projections of 1 and -1. The total number of such vectors is 2^n. It is convenient to numerate all vectors in the following way. The components of vector $|m\rangle$ are obtained by transferring the decimal number m into the binary one and then changing all zeros to minus ones. If we call vectors from 0 to $2^{n-1}-1$ the "negative" set and those from 2^{n-1} to 2^n-1 the "positive" set, then one can easily see that the two sets differ only in sign. In general, $|m\rangle = -|2^n - m - 1\rangle$ for $m = 0,\ 1,\ \dots,\ 2^{n-1}-1$. In the first test (see Fig. 1) N patterns were randomly chosen from the "negative" set, in order to exclude identical patterns, and a hundred of "unknown" vectors were randomly chosen from all the 2^n vectors. In the second test (Fig. 2), instead of choosing "unknown" vectors randomly, the patterns were simply fuzzed with a 10% noise, i.e. ten coordinates chosen at random were inverted for a given pattern.

The results of the two tests are presented in the graphs below.

One can see the general tendency that with the increase of the order of non-linearity k, the informational capacity is increased significantly.

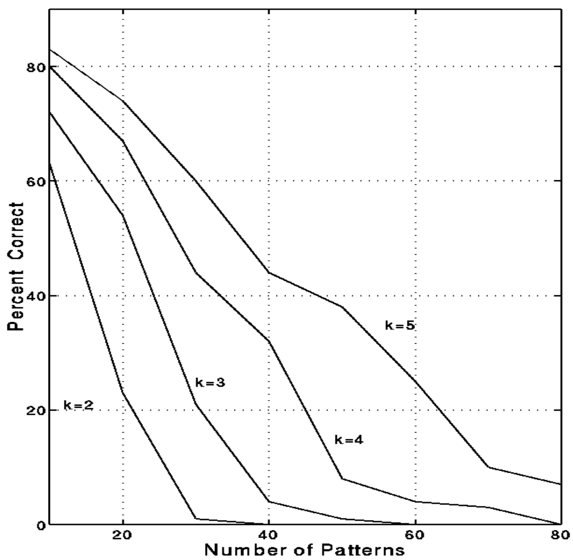

Figure 1. Typical recognition curves for random images.

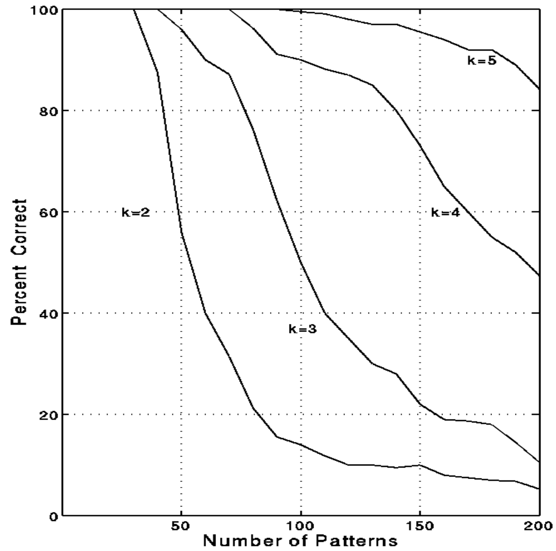

Figure 2. Typical recognition curves for the patterns noised with 10% .

It should be emphasized that the results of the experiments, shown in Figures 1 and 2, are not to be treated as absolute, since the patterns were chosen at random. If one takes other samples of patterns the results may deviate somewhat from those shown above. However, the general tendency of the enhancement of the informational capacity with the increase of k will still be observed.

4. Discussion

The tendency of enhancement of the informational capacity with the increase of the order of non-linearity k can be explained by the following way. In the linear case of k=1, which is the case of the Hopfield net, the maximum number of patterns which can be memorized reliably is naturally restricted by the size of a pattern n, since if we add more patterns then the determinant $\det\left(\left\langle X_i \middle| X_j \right\rangle^k\right)$ will be equal to zero and the completeness of the orthogonal set $\left|V_p^k\right\rangle$ will be broken, whereas the higher cases of will increase this number approximately as n^k. Suppose we have several "bad" patterns which are or almost linearly dependent. It is clear that for higher-order determinants there would be more chance to remain finite. Therefore the multi-particle nets are more sensitive to the difference between the "difficult" patterns.

In conclusion, it should be emphasized the importance of the offered method (2.5) for calculating output images. The importance lies in the storage capacity of data. The most powerful modern computers would hardly be capable of calculating multi-particle nets for the values of k higher than, e.g. four or five, when the actual construction of k-vector is involved. In contrast, our method allows such calculation for **any** values of k, no matter how high. The storage capacity required in this case, in addition to the patterns themselves, is only N^2 values of cofactors.

5. Appendix

A. Orthogonalization procedure

For $\left|F_1^k\right\rangle$ we have $\left|V_1^k\right\rangle = \dfrac{\left|F_1^k\right\rangle}{\left\langle F_1^k \middle| F_1^k \right\rangle}$.

Then we add the pattern $\left|F_2^k\right\rangle$ and write a vector orthogonal to $\left|F_1^k\right\rangle$:

$\left|W_2^k\right\rangle = \left|F_2^k\right\rangle - \left|V_1^k\right\rangle\left\langle F_1^k \middle| F_2^k \right\rangle$ and normalize it $\left|V_2^k\right\rangle = \dfrac{\left|W_2^k\right\rangle}{\left\langle F_2^k \middle| W_2^k \right\rangle}$.

Now we must redefine the vector $\left|V_1^k\right\rangle$ so that it also became orthogonal to $\left|F_2^k\right\rangle$:

$\left|V_1^k\right\rangle \to \left|V_1^k\right\rangle - \left|V_2^k\right\rangle\left\langle F_2^k \middle| V_1^k\right\rangle$.

When we add the p-*th* pattern, we write the vector orthogonal to all of the patterns from $\left|F_1^k\right\rangle$ to $\left|F_{p-1}^k\right\rangle$

$\left|W_p^k\right\rangle = \left|F_p^k\right\rangle - \sum_{q=1}^{p-1}\left|V_q^k\right\rangle\left\langle F_q^k \middle| F_p^k\right\rangle$, normalize it $\left|V_p^k\right\rangle = \dfrac{\left|W_p^k\right\rangle}{\left\langle F_p^k \middle| W_p^k\right\rangle}$, and redefine the vectors :

$\left|V_q^k\right\rangle \to \left|V_q^k\right\rangle - \left|V_p^k\right\rangle\left\langle F_p^k \middle| V_q^k\right\rangle$ for q=1, 2, ... p-1.

When this procedure is performed we obtain the needed set of k-particle vectors $\left|V_p^k\right\rangle$ which can be expressed in the form :

$$\left|V_p^k\right\rangle = \frac{1}{\det\left(\left\langle F_i^k \middle| F_j^k\right\rangle\right)} \sum_{q=1}^{N} M_{pq}^k \left|F_q^k\right\rangle$$

where M_{pq}^k is $N \times N$ matrix composed of cofactors of the matrix $\left(\left\langle F_i^k \middle| F_j^k\right\rangle\right)$.

B. Theorem

For any two vectors $|a\rangle$ and $|b\rangle$, their k-particle scalar product can be found by raising their one-particle scalar product into the k-*th* power:

$$\left\langle F_a^k \middle| F_b^k\right\rangle = \sum_{i_1=1}^{N}\sum_{i_2=1}^{N}\ldots\sum_{i_k=1}^{N}\langle a|i_1\rangle\ldots\langle a|i_k\rangle\langle i_1|b\rangle\ldots\langle i_k|b\rangle = \langle a|b\rangle^k$$

References

1. Hopfield J.J. Neural Networks and physical systems with emergent collective computational abilities. *Proc. Nat. Sci. USA*. 1982. V.79. PP. 2554-2558.

2. Gorban A.N. Training Neural Networks. : publ. USSR-USA JV "ParaGraph", 1990. 160 p. (Russian) (English Translation: *AMSE Transaction, Scientific Siberian,* A, 1993. Vol. 6. Neurocomputing. PP. 1-134).