Decorrelating Parametrical Neural Network

Boris V. Kryzhanovsky
Institute of Optical Neural Technologies
Russian Academy of Sciences
Moscow, Russia
E-mail: iont@iont.ru

Vladimir M. Kryzhanovsky
Institute of Optical Neural Technologies
Russian Academy of Sciences
Moscow, Russia
E-mail: iont@iont.ru

Anatoly B. Fonarev
City University of New York
Department Engineering and Science,
2800 Victory Blvd. SI,
NY 10314

Abstract — We developed a new network architecture, allowing us to increase the recognizing characteristics of the Hopfield Model substantively. In addition it is effective when recognizing correlated patterns. It is shown that the storage capacity of the network increases exponentially with increase of the free parameter of the problem. The boundaries restricting the increase of the storage capacity are defined.

I. INTRODUCTION

The storage capacity of the Hopfield model (HM) is rather small. It allows one to store \( M_{HM} \sim N(1-2p)^2/(2\ln N) \) randomized patterns only, where \( p \in [0, 1/2) \) is a permissible level of distortions. In fact, HM works as an associative memory only if the level of distortions does not exceed 30%, \( p \leq 0.3 \). These estimates are correct for randomized patterns, whose binary coordinates are independent equiprobably distributed random variables. If there are correlations between patterns, the recognizing ability of HM decreases drastically.

For a long time it was considered that the only way to overcome these difficulties was the sparse coding [1], [2], [3]. This technique consists of random diluting of informative coordinates by a great number of spurious coordinates. Then with the aid of a special choice of the threshold and the level of the activity of the patterns the storage capacity can be increased up to maximal value \( M \sim (N/\ln N)^2 \). The last estimate is of theoretical interest only, since no distortions of the patterns are assumed. In the presence of distortions the sparse coding allows one to increase the storage capacity of HM by 1.5-2 times only.

In [4], [5], [6] we suggested another way to increase the binary storage capacity. We based our approach on using the parametrical neural network (PNN) [7]-[11]. PNN is a generalization of HM to the case of \( q \)-nary neurons: the number of different states \( q \) of neurons is greater than two, \( q > 2 \). PNN is the vector version of the neural network of the Potts-glass type [12]. At present PNN possesses the best storage capacity and noise immunity. The storage capacity of PNN is \( q^2 \) times greater than the same characteristics of HM: \( M_{PNN} \sim q^2 M_{HM} \). Its noise immunity is improved significantly too. Basing on these properties of PNN we created an effective neural network architecture that increased the binary storage capacity significantly. We called it the decorrelating PNN, since it deals equally well with both randomized and correlated binary patterns.

The principle of decorrelating PNN operation is as follows. At first the binary patterns are one-to-one mapped into an internal representation using \( q \)-nary neurons. As a result we obtain a set of \( q \)-nary patterns, which are used for PNN construction. The properties of the mapping allow us, first, to eliminate correlations between patterns. Second, the number \( q \) of different states of \( q \)-nary neurons increases exponentially with the mapping parameter \( k \): \( q = 2^k \). Under this approach, the algorithm of binary patterns recognition consists of three stages. At first, the input binary vector is mapped into the \( q \)-nary representation. Then with the aid of PNN its recognition occurs: it is identified with one of the \( q \)-nary patterns. At last, the inverse mapping of the \( q \)-nary pattern into the binary representation takes place.

Since \( q^2 = 2^{2k} \), in the first publications [4]-[6] it was declared that the storage capacity of decorrelating PNN increases exponentially when the parameter \( k \) increases. However, more detailed analysis shows that the increase in storage capacity has its boundary. This boundary depends on the noise level in the system. However, as long as HM works as an associative memory, the boundary value of the storage capacity significantly exceeds \((N/\ln N)^2\), which is the maximal characteristic value for the sparse coding.

In the present report we give rigorous estimates of recognizing characteristics of decorrelating PNN and explain the basic mechanisms of it functioning. The organization of the paper is as follows. In Sect. II we give a short description of PNN and its recognizing characteristics. In Sect. III we describe the algorithm of mapping of binary patterns into \( q \)-nary representation. In Sect. IV we obtain the recognizing characteristics of the decorrelating PNN.

II. PARAMETRICAL NEURAL NETWORK

Here we describe parametrical neural network (PNN) in terms of vector formalism. In order to describe \( q \) different states of neurons we use the set of basis vectors \( e_l \) in the space \( R^q \), \( q \geq 1 \),

\[
e_l = \begin{pmatrix} 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{pmatrix}, \quad l = 1, \ldots, q.
\]
The state of the \(i\)th neuron is described by a column-vector \(x_i\),
\[
  x_i = x_i e_i, \quad x_i = \pm 1, \quad e_i \in \mathbb{R}^q, \quad \left\{ \begin{array}{ll}
    1 \leq i \leq q; \\
    i = 1, \ldots, N.
  \end{array} \right.
\]
The factor \(x_i\) is called the amplitude.

The state of the network as a whole, \(X\), is determined by a set of \(N\) \(q\)-dimensional column vectors \(x_i\): \(X = (x_1, \ldots, x_N)\).

The stored patterns are
\[
  X^{(\mu)} = (x_1^{(\mu)}, x_2^{(\mu)}, \ldots, x_N^{(\mu)}),
\]
where \(x_i^{(\mu)} = x_i(\mu) e_i^{(\mu)}, \quad x_i^{(\mu)} = \pm 1, \quad 1 \leq i^{(\mu)} \leq q, \quad \mu = 1, \ldots, M,
\]
and the local field is
\[
  h_i = \frac{1}{N} \sum_{j=1}^{N} T_{ij} x_j.
\]
The \((q \times q)\)-matrix \(T_{ij}\) describes the interconnection between the \(i\)th and the \(j\)th vector-neurons. In the generalized Hebb form it is defined with the aid of the tensor product,
\[
  T_{ij} = (1 - \delta_{ij}) \sum_{\mu=1}^{M} x_i^{(\mu)} x_j^{(\mu)}\text{,} \quad i, j = 1, \ldots, N,
\]
where \(\delta_{ij}\) is the Kronecker symbol and \(x_j^{(\mu)}\) is the \(q\)-dimensional row vector.

The local field that acts on \(i\)th neuron of the state \(X(t) = (x_1(t), x_2(t), \ldots, x_N(t))\) can be written as
\[
  h_i(t) = \sum_{l=1}^{q} A_{i}^{l} e_l,
\]
where
\[
  A_{i}^{l} = \sum_{j \neq i} \sum_{\mu=1}^{M} (e_l x_j^{(\mu)})(x_j^{(\mu)} x_j(t)).
\]
The dynamics of the system is defined in the natural way: under action of the local field \(h_i\) the \(i\)th vector-neuron gets orientation that is as close to the local field direction as possible. In other words, the state of the \(i\)th neuron in the next time step, \(t+1\), is determined by the rule:
\[
  x_i(t+1) = x_{\text{max}} e_{\text{max}}, \quad x_{\text{max}} = \text{sgn} \left( A_{i}^{l} \right),
\]
where the subscript \(\text{max}\) denotes the greatest in modulus amplitude \(A_{i}^{l}\) in (1). The evolution of the system consists of consequent changes of orientations of vector-neurons according to the rule (2). Sooner or later the network finds itself at a fixed point.

Let us see how efficiently PNN recognizes noisy patterns. Suppose the distorted \(m\)th pattern \(X^{(m)}\) comes to the system input, i.e. the neurons in the initial states are described as \(x_i = a_i b_i x_i^{(m)}\). Here \(a_i\) is the multiplicative noise operator, which changes the sign of the amplitude \(x_i^{(m)}\) with a probability \(a\) and keeps it the same with the probability \(1-a\). The operator \(b_i\) changes the basis vector \(e_i^{(m)}\) by any other one with a probability \(b\) and retains it unchanged with the probability \(1-b\). In other words, \(a\) is the probability of an error in the sign of the amplitude of the neuron, \(b\) is the probability of an error in the vector state of the neuron. The network recognizes the reference pattern \(X^{(m)}\) correctly, if the output of the \(i\)th neuron defined by Eq.(2) is equal to \(x_i^{(m)}\). Otherwise PNN fails to recognize the pattern \(X^{(m)}\). Let us estimate the probability of the error of the \(m\)th pattern recognition.

We separate the local field into a signal- and a noise-terms
\[
  (x_i^{(m)}, h_i) = \xi + \eta = \sum_{j \neq i}^{N} \sum_{\mu=1}^{M} \sum_{\mu \neq m}^{M} \eta_j^{(\mu)}
\]
(for details see [11]). The quantity \(\xi\) is the useful signal. It is defined by influence of the \(m\)th pattern onto the \(i\)th neuron. The quantity \(\eta\) symbolizes the inner noise, connected with noisy influence of all other patterns. Random variables \(\xi, \eta\) are asymptotically normal distributed. The distribution parameters are
\[
  E(\xi) \sim N(1-2a)(1-b), \quad D(\xi) \to 0, \quad D(\eta) \sim \frac{M N}{q^2}.
\]

Now the probability of the recognition error of a coordinate \(x_i^{(m)}\) can be calculated by integration of the normally distributed \(\eta\) in the region where \(\eta > E(\xi)\). If now we consider not one vector-coordinate, but the whole pattern and use the standard approximation ([10],[11]), the estimate for the probability of the recognition error of the pattern \(X^{(m)}\) is
\[
  \Pr_{err} \sim N(\frac{2q-1}{\beta} \exp(-\beta^2/2), \quad \text{where} \quad \beta = \frac{N(1-b)^2(1-2a)^2}{M} q^2.
\]

Then, the asymptotically possible value of the storage capacity of PNN is
\[
  M_{PNN} = \frac{N(1-b)^2(1-2a)^2}{2 \ln(Nq)} \cdot q^2.
\]

When \(q = 1\), Eqs.(3)-(4) transform into the well-known results for HM (in this case \(b = 0\)). When \(q\) increases, the probability of the error (3) decreases exponentially, i.e. the noise immunity of PNN increases noticeably. In the same time the storage capacity of the network increases proportionally to \(q^2\). For example, let us set \(\Pr_{err} = 0.01\). In the framework of HM, with this probability of the error we can recognize any of \(M = N/10\) patterns, each of which is less then 30% noisy. In the same time, PNN with \(q = 64\) allows us to recognize any of \(M = 5N\) patterns with 90% noise, or any of \(M = 50N\) patterns with 65% noise. We use these outstanding characteristics of PNN to improve the binary storage capacity significantly.

III. MAPPING ALGORITHM

Here we describe the mapping algorithm of binary patterns into \(n\)-ary ones. Later on the \(n\)-ary patterns are used for PNN construction.

Let \(Y = (y_1, y_2, \ldots, y_N)\) be \(N\)-dimensional binary vector, \(y_i = \{-1/1\}\). We divide it mentally into \(n\) fragments of \(k+1\) elements each, \(N = n(k+1)\). With each fragment we associate
an integer number ±l according the following rule: i) the first element of the fragment defines the sign of the number; ii) the other k elements of the fragment determine the absolute value of the number l,

\[ l = 1 + \sum_{i=2}^{k+1} (y_i + 1) \cdot 2^{k-i}; \quad 1 \leq l \leq 2^k. \]

(In other words, we interpret the last k elements of the fragment as the binary notation of the integer l.)

After that, we associate each fragment with a vector \( x = \pm e_l \), where \( e_l \) is the /l unit vector in the space \( R^n \), and \( q = 2^k \).

We see that any binary vector \( Y \in R^N \) one-to-one corresponds to a set of \( n \) \( q \)-dimensional unit vectors, \( X = (x_1, x_2, \ldots, x_n) \), which we call the internal image of the binary vector \( Y \). (In the next Section we use the internal images \( X \) for PNN construction.) The number \( k \) is called a mapping parameter.

For example, the binary vector \( Y = (-1, 1, -1, 1, -1, 1, -1, 1) \) can be split into two fragments of four elements: \((-1, 1, -1, 1) \) and \((1, -1, 1, -1) \); the mapping parameter \( k \) is equal 3, \( k = 3 \). The first fragment \((-5 \text{ in our notations}) \) corresponds to the vector \(-e_3 \) from the space of dimensionality \( q = 2^3 = 8 \), and the second fragment \((+2 \text{ in our notations}) \) corresponds to the vector \( e_2 \in R^8 \). The relevant mapping can be written as \( Y \rightarrow X = (-e_3, +e_2) \).

It is important that the mapping is biunique, i.e., the binary vector \( Y \) can be restored uniquely from its internal image \( X \). It is even more important that the mapping eliminates correlations between internal images. For example, suppose we have two 75% overlapping binary vectors

\[ \begin{align*}
Y_1 & = (1, -1, -1, -1, -1, -1, -1, 1) \\
Y_2 & = (1, -1, -1, 1, 1, -1, -1, 1)
\end{align*} \]

Let us divide each vector into four fragments of two elements each. In other words, we map these vectors with the mapping parameter \( k = 1 \). As a result we obtain two internal images \( X_1 = (+e_1, -e_1, -e_1, -e_2) \) and \( X_2 = (+e_1, -e_2, +e_1, -e_2) \) with \( e_l \in R^2 \). The overlapping of these images is 50%. If the mapping parameter \( k = 3 \) is used, the relevant images \( X_1 = (+e_1, -e_2) \) and \( X_2 = (+e_2, +e_2) \) with \( e_l \in R^8 \) do not overlap at all.

IV. DECORRELATING PNN

In this Section we describe the work of our model as a whole, i.e. the mapping of original binary patterns into internal images and recognition of these images with the aid of PNN.

For a given mapping parameter \( k \) we apply the procedure from Sect. III to a set of binary patterns \( \{Y^{(\mu)}\} \in R^N, \mu \in [1, \mu] \). As a result we obtain a set of internal images \( \{X^{(\mu)}\} \) with \( n = N/(k+1) \) \( q \)-dimensional vector-neurons, where \( q = 2^k \). These images can be considered as randomized ones.

With the aid of these images we build PNN as it was described in Sect. II. Let us estimate the number of patterns, which can be stored by this network in the absence of distortions. For this purpose we substitute \( n \) and \( q \) into the expression (4), where we set \( a = b = 0 \):

\[ M \sim \frac{N}{2^{N \ln N}} \cdot \frac{2^{2k}}{k(1+k/\ln N)} = M_{HM}, \quad q^{2k}/k(1+k/\ln N). \]  \( (5) \)

We see that the storage capacity exponentially increases with \( k \). However, \( k \) cannot increase unrestrictedly. Its limiting value is defined by the requirement of number of vector-neurons \( n \) being sufficiently large. Only in this case the estimates (3) and (4) obtained with the aid the central limit theorem are correct. Thus, we need

\[ n = \frac{N}{k+1} \gg 1 \Rightarrow k << N. \]  \( (6) \)

For example, when \( N \sim 10^3 \) and \( k = 10 \), it can be assumed that the condition (6) is fulfilled. Then we have \( M \sim 10^4 \cdot M_{HM} \). In this case the storage capacity of the decorrelating PNN exceeds the storage capacity of HM by four orders of magnitude.

Now let us estimate the number of patterns, which can be recognized by the decorrelating PNN in the case of distorted input binary vectors. Let the probability of distortions of coordinates of input binary patterns be \( p < 1/2 \). Then mapping the vector \( Y^{(m)} \) into \( q \)-nary representation (see Sect. III), we obtain the distorted pattern

\[ \tilde{X}^{(m)} = (a_1 \hat{b}_1 x_1^{(m)}, a_2 \hat{b}_2 x_2^{(m)}, \ldots, a_N \hat{b}_N x_N^{(m)}). \]

It is easy to see that \( a = p, b = 1 - (1-p)^k \). Substituting these expressions into Eqs.(4), we obtain

\[ M \sim \frac{N(1-2p)^2}{2^{N \ln N}} \cdot \frac{(2(1-p))^{2k}}{k(1+k/\ln N)}. \]  \( (7) \)

The first factor in the right-hand side of Eq.(7) is the storage capacity of HM in the presence of distortions. Since always the inequality \( 2(1-p) > 1 \) is fulfilled, the second factor describes the exponential growth of the storage capacity with \( k \). For sufficiently large \( k \), the substantial superiority of the decorrelating PNN comparing with HM can be gained.

Note, the value of \( k \) cannot be increased unlimitedly. In addition to the restriction (6), it is necessary to have at least one undistorted vector-neuron among the vector-neurons of the distorted pattern \( \tilde{X}^{(m)} \). Otherwise PNN simply cannot recognize the correct values of the vector-coordinates of the pattern. It is easy to see that the number of undistorted vector-neurons of \( \tilde{X}^{(m)} \) is equal to \( n(1-p)^{k+1} \), and it has to be not less than 2:

\[ n(1-p)^{k+1} \geq 2 \Rightarrow (1-p)^{k+1} \geq \frac{2(k+1)}{N}. \]  \( (8) \)

Equations (6) and (8) define the boundaries restricting the increase of the binary storage capacity obtained by means of the decorrelating PNN.

For the following analysis it is convenient to rewrite Eq.(7) in the form

\[ M \sim N^R \cdot \frac{(1-2p)^2}{2(2k+1) \ln N}. \]
where
\[ R = 1 + 2k \ln \left( \frac{2(1 - p)}{\ln N} \right). \quad (9) \]

We see, that if noises are absent \((p = 0)\), already if \(k > \ln N/2\), the storage capacity of the decorrelating PNN becomes larger than \(N^2\), which is the characteristic value of the storage capacity for the sparse coding (see Introduction).

In the presence of a noise \((p \neq 0)\), the value of the parameter \(k\) cannot exceed the critical value
\[ k_c = \frac{1}{2p \ln N}. \]

When \(k\) increases inside the region \(0 \leq k < k_c\), the storage capacity \(M\) increases exponentially. However, when \(k \geq k_c\), the network failed to recognize patterns. In this case, the basin of attraction becomes so narrow, that a distorted pattern falls out of its boundaries.

\[ p_c = \frac{1}{2k \ln N}. \]

In Fig.2 the dependence of the size of the basin of attraction on the value of the parameter \(k\) is shown. The size of the basin of attraction decreases when \(k\) increases. The binary patterns with the level of distortions \(p > p_c\) would not be recognized.

**ACKNOWLEDGMENT**

The authors would like to thank Dr. Leonid Litinskii for helpful discussions.

The work was supported by the program "Intellectual Computer Systems" (the project 2.45) and in part by Russian Basic Research Foundation (grants 04-07-90038 and 03-01-00355).

**REFERENCES**