Analysis and Design of Associative Memories Based on Cellular Neural Networks with Space-invariant Cloning Templates

Zhigang Zeng and Jun Wang

Abstract—This paper presents a new design procedure for the synthesis of associative memories based on cellular neural networks characterized by input and output matrices obtained based on cloning templates by solving a set of inequalities. The design parameters are few even though the dimension of patterns may be very high. The design procedure enables heteroassociative or autoassociative memories to be synthesized with assured global exponential stability and feeding retrieval probes via external inputs rather than initial conditions. Two specific examples are shown to illustrate the applicability of the methodology.

I. INTRODUCTION

Associative memories are brain-style storages designed to memorize a set of patterns as stable equilibria such that the stored patterns can be reliably retrieved with the initial probes containing sufficient information about the patterns.

In past two decades, several methods have been proposed for designing associative memories using neural networks such as cellular neural networks (CNNs) [1]-[16]. Generally, there exist two approaches to pattern recalling in associative memories. In the first method, neural networks such as CNNs with multiple equilibrium points are directly regarded as associative memories [1]-[14]. In this case, a recalling probe, which is sufficiently close to the pattern to be retrieved, is set as initial states and the state variables converge to one of the locally asymptotically stable equilibria which corresponds to the pattern to be retrieved. In the second method, by assuring that each trajectory globally converges to a unique equilibrium point which depends on the external input instead of initial state, neural networks such as CNNs with unique equilibrium points enable both heteroassociative and autoassociative patterns to be recalled [15], [16], [17].

In recently years, it is found that any n-neuron cellular neural networks with delays (DCNN) have up to \(2^n\) locally stable equilibria [18], which could be used as large-capacity associative memories. When the initial state of a neural network is located at an given attractive region, the corresponding equilibrium point corresponds to an associatively memorized pattern. In multi-equilibrium associative memories, it is necessary that each dynamic trajectory converges to one of locally asymptotically stable equilibrium points. Because it is difficult to avoid spurious equilibria, accurate pattern recalling can not be guaranteed by using the first method. In contrast, there exists no spurious equilibrium point if an associative memory is properly designed using the second method. Hence, the second design method may be more desirable.

Despite of numerous studies on associative memories, many design issues demand further investigations. For example, it is difficult to obtain parameter matrices and space-invariant cloning templates that ensures various patterns can be associatively memorized. In addition, the robustness of input patterns is rarely considered. It is desirable to design robust associative memories to withstand the parameter perturbations and random disturbance on patterns.

Furthermore, in hardware implementation of neural networks, time delays even time-varying delays in neuron signal transmission or processing are often inevitable. It is more realistic to design neural networks which are robust on delays. When an associative memory is designed as a CNN, it may end up with DCNN after implementation. Therefore the analysis of DCNN is more realistic.

The dynamic rule of a CNN can be completely specified by its cloning template. One of the advantages of considering a space-invariant cloning template for CNNs is that the small number of connection weights are required for describing a CNN which can simplify hardware implementation [13].

In this paper, a design methodology for DCNNs described using space-invariant cloning templates is developed based on the stability results in [19]-[20]. By using the proposed approach, when a prespecified probe is fed to the designed DCNNs in the allowable region of the bias vector, the stable output variables convergence to a fixed point that correspond to the desired pattern. The design procedure enables input matrix and output matrix to be determined based on cloning templates with a few design parameters.

The remaining paper consists of five sections. Section II describes some preliminaries. In Section III, several sufficient conditions are obtained that guarantee DCNNs with invariant templates to have a unique stable equilibrium point corresponding to a memory pattern. In Section IV, a design procedure for heteroassociative memories based on DCNNs are presented. In Section V, two illustrative examples are given to demonstrate the use of the proposed approach. Finally, concluding remarks are made in Section VI.
II. PRELIMINARIES

A. Design Problem

Denote \{-1, 1\}^n as the set of n-dimensional bipolar vectors; i.e.,
\[
\{-1, 1\}^n = \{x \in \mathbb{R}^n, x = (x_1, x_2, \ldots, x_n)^T, x_i = 1 \text{ or } -1, i = 1, 2, \ldots, n\},
\]
\(-1, 1\)^n \times \{-1, 1\}^m as the product of the set of n-dimensional and m-dimensional bipolar vectors; i.e.,
\[
\{-1, 1\}^n \times \{-1, 1\}^m = \{(x, y) \in \mathbb{R}^{n+m}, x = (x_1, x_2, \ldots, x_n)^T, x_i = 1 \text{ or } -1, i = 1, 2, \ldots, n, y = (y_1, \ldots, y_m)^T, y_j = 1 \text{ or } -1, j = 1, 2, \ldots, m\}.
\]

Hence, \{-1, 1\}^n \times \{-1, 1\}^m is made up of 2n+m elements. We have the following design problem:

**Design Problem.** Given \(p\) (\(p \leq \min\{2^n, 2^m\}\)) pairwise vectors (paired patterns and probes) \(s\{1\}, u\{1\}\), \(s\{2\}, u\{2\}\), \ldots, \(s\{p\}, u\{p\}\), \(s\{\ell\}, u\{\ell\} \in \{-1, 1\}^n \times \{-1, 1\}^m, \ell \in \{1, 2, \ldots, p\}\) design an associative memory based on a neural network such that if \(u^{(\ell)}\) \((\ell \in \{1, 2, \ldots, p\}\) is fed into the associative memory from its input as a probe, then the output vector of neural network converges to corresponding pattern \(s^{(\ell)}\). When \(s^{(\ell)} \approx u^{(\ell)}\), \(s^{(\ell)}\) is said to be autoassociatively memorized with \(u^{(\ell)}\) in the associative memory. Otherwise, \(s^{(\ell)}\) is said to be heteroassociatively memorized with \(u^{(\ell)}\).

B. DCNNs Described Using Cloning Templates

Consider a DCNN described by using space-invariant templates where the cells are arranged in a two-dimensional rectangular array composed of \(N\) rows and \(M\) columns, and the inputs are arranged in a two-dimensional rectangular array composed of \(N\) rows and \(M\) columns.

The dynamics of such a DCNN is governed by the following normalized equations:
\[
\frac{dx_{ij}(t)}{dt} = -x_{ij}(t) + \sum_{k=1}^{k_2(i, r_1)} \sum_{l=1}^{l_2(j, r_2)} \left( \hat{a}_{k,l} f(x_{i+k,l+j}(t)) \right) + \sum_{k=1}^{k_2(i, r_1)} \sum_{l=1}^{l_2(j, r_2)} \left( \hat{b}_{k,l} f(y_{i+k,l+j}(t) + \tau_{ij}(t)) \right)
\]
\[
+ \sum_{k=1}^{k_2(i, r_1)} \sum_{l=1}^{l_2(j, r_2)} \left( \hat{a}_{k,l} f(x_{i+k,l+j}(t)) \right) + \sum_{k=1}^{k_2(i, r_1)} \sum_{l=1}^{l_2(j, r_2)} \left( \hat{b}_{k,l} f(y_{i+k,l+j}(t) + \tau_{ij}(t)) \right)
\]
\[
+ \sum_{k=1}^{k_2(i, r_1)} \sum_{l=1}^{l_2(j, r_2)} \left( \hat{a}_{k,l} f(x_{i+k,l+j}(t)) \right) + \sum_{k=1}^{k_2(i, r_1)} \sum_{l=1}^{l_2(j, r_2)} \left( \hat{b}_{k,l} f(y_{i+k,l+j}(t) + \tau_{ij}(t)) \right)
\]
\[
\hat{a}_{k,l} f(x_{i+k,l+j}(t)) + \hat{b}_{k,l} f(y_{i+k,l+j}(t) + \tau_{ij}(t))
\]
\[
\frac{dx_{ij}(t)}{dt} = -x_{ij}(t) + Af(x(t)) + Bf(x(t) - \tau(t)) + Du + v,
\]

(1)

where the coefficient matrices \(A, B, D\) and \(D\) are obtained through the templates \(A, B, D\). For \(i = 1, 2, \ldots, N\) and \(j = 1, 2, \ldots, M\), let \(u_{i+M,j+i} = \hat{u}_{ij}, v_{i+M,j+i} = \hat{v}_{ij}\); the vectors \(u\) and \(v\) are obtained through \(\hat{u}_{ij}\) and \(\hat{v}_{ij}\), respectively, vector-valued activation functions \(f(x(t)) = (f(x_1(t)), f(x_2(t)), \ldots, f(x_{NM}(t)))^T\), \(f(x(t) - \tau(t)) = (f(x_1(t) - \tau_1(t)), f(x_2(t) - \tau_2(t)), \ldots, f(x_{NM}(t) - \tau_{NM}(t)))^T\); the delay \(\tau_{ij}(t)\) is obtained through \(\hat{\tau}_{ij}(t)\).

Let \((t_0, \tau_0, D)\) be the space of continuous functions mapping \([t_0, \tau_0, D)\) to \(D \in \mathbb{R}^{NM}\) with the norm defined by \(|x| = \max_{1 \leq i \leq N, 1 \leq j \leq M} \sup_{[t_0, \tau_0, D]} |\phi_u(x(t))|\), where \(\phi(x) = (\phi_1(x), \phi_2(x), \ldots, \phi_{NM}(x))^T\). Denote \(\phi(\hat{\theta}) = (\phi_1(\hat{\theta}), \phi_2(\hat{\theta}), \ldots, \phi_{NM}(\hat{\theta}))^T\),

(4)

as the initial condition of DCNN (1), where \(\phi(\hat{\theta}) \in C([t_0, \tau_0, D])\). Denote \(x(t; t_0, \hat{\theta})\) as the solution of (1) with initial condition (4). It means that \(x(t; t_0, \hat{\theta})\) is continuous and satisfies (1) and \(x(s; t_0, \hat{\theta}) = \phi(s), \text{ for s in } [t_0, \tau_0]\). The matrices \(A, B, D\) in DCNN (3) depend on the established order among the cells and on the cloning templates and the delay cloning template. For example, for a two-dimensional space-invariant DCNN with a neighborhood radius \(r\), its cloning template \(\hat{A}\) is \((2r+1) \times (2r+1)\) real matrix; i.e.,

\[
\hat{A} = \begin{bmatrix}
    a_{-r,-r} & \cdots & a_{-r,0} & \cdots & a_{-r,r} \\
    \vdots & \ddots & \vdots & \ddots & \vdots \\
    a_{0,-r} & \cdots & a_{0,0} & \cdots & a_{0,r} \\
    \vdots & \ddots & \vdots & \ddots & \vdots \\
    a_{r,-r} & \cdots & a_{r,0} & \cdots & a_{r,r}
\end{bmatrix}
\]

For simplicity, as required in most applications, we always assume that \(r_1 = r_2 = \hat{r}_1 = \hat{r}_2 = 1\) hereafter. Hence, the
cloning templates are
\[
A = \begin{bmatrix}
\hat{a}_{-1,-1} & \hat{a}_{-1,0} & \hat{a}_{-1,1} \\
\hat{a}_{0,-1} & \hat{a}_{0,0} & \hat{a}_{0,1} \\
\hat{a}_{1,-1} & \hat{a}_{1,0} & \hat{a}_{1,1}
\end{bmatrix},
\]
\[
B = \begin{bmatrix}
\hat{b}_{-1,-1} & \hat{b}_{-1,0} & \hat{b}_{-1,1} \\
\hat{b}_{0,-1} & \hat{b}_{0,0} & \hat{b}_{0,1} \\
\hat{b}_{1,-1} & \hat{b}_{1,0} & \hat{b}_{1,1}
\end{bmatrix},
\]
and
\[
D = \begin{bmatrix}
\hat{d}_{-1,-1} & \hat{d}_{-1,0} & \hat{d}_{-1,1} \\
\hat{d}_{0,-1} & \hat{d}_{0,0} & \hat{d}_{0,1} \\
\hat{d}_{1,-1} & \hat{d}_{1,0} & \hat{d}_{1,1}
\end{bmatrix},
\]
The matrix \( A \) in (3) is composed of the template has the form
\[
A = \begin{bmatrix}
A(1) & A(2) & \cdots & 0 \\
A(3) & A(1) & \cdots & 0 \\
0 & A(3) & \cdots & 0 \\
0 & 0 & \cdots & A(3)
\end{bmatrix}_{(MN) \times (MN)},
\]
where
\[
A(1) = \begin{bmatrix}
\hat{a}_{0,0} & \hat{a}_{0,1} & 0 & \cdots & 0 & 0 \\
\hat{a}_{0,-1} & \hat{a}_{0,0} & \hat{a}_{0,1} & \cdots & 0 & 0 \\
0 & \hat{a}_{0,-1} & \hat{a}_{0,0} & \cdots & 0 & 0 \\
0 & 0 & \hat{a}_{0,-1} & \cdots & 0 & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
0 & 0 & 0 & \cdots & \hat{a}_{0,0} & \hat{a}_{0,1} \\
0 & 0 & 0 & \cdots & \hat{a}_{0,-1} & \hat{a}_{0,0}
\end{bmatrix}_{M \times M},
\]
\[
A(2) = \begin{bmatrix}
\hat{a}_{1,0} & \hat{a}_{1,1} & 0 & \cdots & 0 & 0 \\
\hat{a}_{1,-1} & \hat{a}_{1,0} & \hat{a}_{1,1} & \cdots & 0 & 0 \\
0 & \hat{a}_{1,-1} & \hat{a}_{1,0} & \cdots & 0 & 0 \\
0 & 0 & \hat{a}_{1,-1} & \cdots & 0 & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
0 & 0 & 0 & \cdots & \hat{a}_{1,0} & \hat{a}_{1,1} \\
0 & 0 & 0 & \cdots & \hat{a}_{1,-1} & \hat{a}_{1,0}
\end{bmatrix}_{M \times M},
\]
\[
A(3) = \begin{bmatrix}
\hat{a}_{-1,0} & \hat{a}_{-1,1} & 0 & \cdots & 0 \\
\hat{a}_{-1,-1} & \hat{a}_{-1,0} & \hat{a}_{-1,1} & \cdots & 0 \\
0 & \hat{a}_{-1,-1} & \hat{a}_{-1,0} & \cdots & 0 \\
0 & 0 & \hat{a}_{-1,-1} & \cdots & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
0 & 0 & 0 & \cdots & \hat{a}_{-1,1} \\
0 & 0 & 0 & \cdots & \hat{a}_{-1,0}
\end{bmatrix}_{M \times M}.
\]

B. \( B(1), B(2), B(3) \) can be similarly defined.

The matrix \( D \) in (3) is similarly composed of the template has the form
\[
D = \begin{bmatrix}
D(1) & D(2) & \cdots & 0 \\
D(3) & D(1) & \cdots & 0 \\
0 & D(3) & \cdots & 0 \\
0 & 0 & \cdots & D(3)
\end{bmatrix}_{(MN) \times (MN)},
\]
where
\[
D(1) = \begin{bmatrix}
\hat{d}_{0,0} & \hat{d}_{0,1} & 0 & \cdots & 0 & 0 \\
\hat{d}_{0,-1} & \hat{d}_{0,0} & \hat{d}_{0,1} & \cdots & 0 & 0 \\
0 & \hat{d}_{0,-1} & \hat{d}_{0,0} & \cdots & 0 & 0 \\
0 & 0 & \hat{d}_{0,-1} & \cdots & 0 & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
0 & 0 & 0 & \cdots & \hat{d}_{0,0} & \hat{d}_{0,1} \\
0 & 0 & 0 & \cdots & \hat{d}_{0,-1} & \hat{d}_{0,0}
\end{bmatrix}_{M \times M},
\]
\[
D(2) = \begin{bmatrix}
\hat{d}_{1,0} & \hat{d}_{1,1} & 0 & \cdots & 0 & 0 \\
\hat{d}_{1,-1} & \hat{d}_{1,0} & \hat{d}_{1,1} & \cdots & 0 & 0 \\
0 & \hat{d}_{1,-1} & \hat{d}_{1,0} & \cdots & 0 & 0 \\
0 & 0 & \hat{d}_{1,-1} & \cdots & 0 & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
0 & 0 & 0 & \cdots & \hat{d}_{1,1} & \hat{d}_{1,0} \\
0 & 0 & 0 & \cdots & \hat{d}_{1,-1} & \hat{d}_{1,0}
\end{bmatrix}_{M \times M},
\]
\[
D(3) = \begin{bmatrix}
\hat{d}_{-1,0} & \hat{d}_{-1,1} & 0 & \cdots & 0 \\
\hat{d}_{-1,-1} & \hat{d}_{-1,0} & \hat{d}_{-1,1} & \cdots & 0 \\
0 & \hat{d}_{-1,-1} & \hat{d}_{-1,0} & \cdots & 0 \\
0 & 0 & \hat{d}_{-1,-1} & \cdots & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
0 & 0 & 0 & \cdots & \hat{d}_{-1,1} \\
0 & 0 & 0 & \cdots & \hat{d}_{-1,0}
\end{bmatrix}_{M \times M}.
\]

III. MAIN RESULTS

A. Notations
For \( \ell \in \{1, 2, \ldots, p\}, k \in \{1, 2, \ldots, N\} \), let
\[
P_k^{(\ell)} = \begin{bmatrix}
0 & u^{(\ell)}(k-1)M + 1 & u^{(\ell)}(k-1)M + 2 \\
u^{(\ell)}(k-1)M + 1 & u^{(\ell)}(k-1)M + 2 & u^{(\ell)}(k-1)M + 3 \\
u^{(\ell)}(k-1)M + 2 & u^{(\ell)}(k-1)M + 3 & u^{(\ell)}(k-1)M + 4 \\
\vdots & \vdots & \vdots \\
u^{(\ell)}(kM - 2) & u^{(\ell)}(kM - 1) & u^{(\ell)}(kM) \\
u^{(\ell)}(kM) & u^{(\ell)}(kM) & 0
\end{bmatrix}_{M \times 3},
\]
\[
P^{(\ell)} = \begin{bmatrix}
0 & P_1^{(\ell)} & P_2^{(\ell)} \\
\vdots & \vdots & \vdots \\
P_{N-2}^{(\ell)} & P_{N-1}^{(\ell)} & P_N^{(\ell)} \\
P_N^{(\ell)} & P_{N-1}^{(\ell)} & 0
\end{bmatrix}_{(M \times N) \times 9}.
\]

For \( \ell \in \{1, 2, \ldots, p\}, k \in \{1, 2, \ldots, N\} \), let
\[
Q_k^{(\ell)} = \begin{bmatrix}
0 & s_{(k-1)M + 1}^{(\ell)} & s_{(k-1)M + 2}^{(\ell)} \\
s_{(k-1)M + 1}^{(\ell)} & s_{(k-1)M + 2}^{(\ell)} & s_{(k-1)M + 3}^{(\ell)} \\
s_{(k-1)M + 2}^{(\ell)} & s_{(k-1)M + 3}^{(\ell)} & s_{(k-1)M + 4}^{(\ell)} \\
\vdots & \vdots & \vdots \\
s_{kM - 2}^{(\ell)} & s_{kM - 1}^{(\ell)} & s_{kM}^{(\ell)} \\
s_{kM - 1}^{(\ell)} & s_{kM}^{(\ell)} & 0
\end{bmatrix}_{M \times 3}.
For autoassociative memories, \( s(t) = u(t) \). Thus \( P(t) = Q(t) \).

Denote \( P(t)(i) \) and \( Q(t)(i) \) as the \( i \)-th row vector of matrices \( P(t) \) and \( Q(t) \), respectively. Denote \( l_{A+B} = (a_{-1,-1} + b_{-1,-1}, a_{-1,0} + b_{-1,0}, a_{-1}, a_{0,0}, a_{0,0,1} + b_{0,0}, a_{1,0}, a_{1,1} + b_{1,1})^T, l_{A+B} = (0, 0, 0, 0, 0, 0, 0, 0)^T, l_D = (d_{-1,-1}, d_{-1,0}, d_{-1}, d_{0,0}, d_{0,1}, d_{1,0}, d_{1,1})^T \). From (3) and the definitions of \( P(t) \) and \( Q(t) \),

\[
(A + B)s(t) = Q(t)l_{A+B},
\]

\[
D_0(t) = P(t)l_D.
\]

Let \( N = \{1, 2, \cdots, n\}; N_0 = N - N_1 - N_2 - N_3 - N_4 - N_5 - N_6 - N_7 - N_8 - N_9 \), where

\[
N_1 = \{i \mid s_1(i) = s_2(i) = \cdots = s_9(i) = 1, \; i \in N \},
\]

\[
N_2 = \{i \mid s_1(i) = s_2(i) = \cdots = s_9(i) = -1, \; i \in N \}.
\]

For \( i \in N_0 \), let

\[
\mathcal{J}_i^+ = \{\ell \mid s_1(i) = 1, \ell \in \{1, 2, \cdots, p\}\},
\]

\[
\mathcal{J}_i^- = \{\ell \mid s_1(i) = -1, \ell \in \{1, 2, \cdots, p\}\}.
\]

For \( i \in N_1 \), let

\[
v_1^+ = \max_{i \in \{1, 2, \cdots, p\}} \{1 - Q(i)(i)l_{A+B} - P(t)(i)l_D\},
\]

\[
v_1^- = +\infty.
\]

For \( i \in N_2 \), let

\[
v_2^+ = -\infty,
\]

\[
v_2^- = \max_{i \in \{1, 2, \cdots, p\}} \{-1 - Q(i)(i)l_{A+B} - P(t)(i)l_D\}.
\]

For \( i \in N_0 \), let

\[
v_0^+ = \max_{i \in \mathcal{J}_i^+} \{1 - Q(i)(i)l_{A+B} - P(t)(i)l_D\},
\]

\[
v_0^- = \max_{i \in \mathcal{J}_i^-} \{-1 - Q(i)(i)l_{A+B} - P(t)(i)l_D\}.
\]

Denote \( |N_0|, |\mathcal{J}_i^+|, |\mathcal{J}_i^-| \) as the number of elements in sets \( N_0, \mathcal{J}_i^+, \mathcal{J}_i^- \), respectively. Obviously, \( 0 \leq |N_0| \leq n, |\mathcal{J}_i^+| + |\mathcal{J}_i^-| = p \). Hence, the sets

\[
\Xi = \{(Q(r)(i) - Q(q)(i), P(r)(i) - P(q)(i))T, \; i \in N_0, r \in \mathcal{J}_i^+, q \in \mathcal{J}_i^- \},
\]

\[
\Xi(s) = \{(Q(r)(i) - Q(q)(i))T, i \in N_0, r \in \mathcal{J}_i^+, q \in \mathcal{J}_i^- \},
\]

\[
\Xi(u) = \{(P(r)(i) - P(q)(i))T, i \in N_0, r \in \mathcal{J}_i^+, q \in \mathcal{J}_i^- \}
\]

are made up of \( \xi = \sum_{i \in N_0} |\mathcal{J}_i^+| \times |\mathcal{J}_i^-| \) vectors. Let \( \Xi_1, \Xi_2, \cdots, \Xi_\xi \in \Xi, \Xi_1(s), \Xi_2(s), \cdots, \Xi_\xi(s) \in \Xi(s), \Xi_1(u), \Xi_2(u), \cdots, \Xi_\xi(u) \in \Xi(u), \) and \( \forall i \neq j, \Xi_i \neq \Xi_j, \Xi_i(s) \neq \Xi_j(s), \Xi_i(u) \neq \Xi_j(u), \) and

\[
\Lambda = (\Xi_1, \Xi_2, \cdots, \Xi_\xi)_{18 \times \xi},
\]

\[
\Lambda(s) = (\Xi_1(s), \Xi_2(s), \cdots, \Xi_\xi(s))_{9 \times \xi},
\]

\[
\Lambda(u) = (\Xi_1(u), \Xi_2(u), \cdots, \Xi_\xi(u))_{9 \times \xi}.
\]

**B. Existence of Cloning Templates**

**Lemma 1** [21]. Let \( A \) be a matrix and \( x \) and \( b \) be vectors. Then the system \( Ax = b, x \geq 0 \) has no solution if and only if the system \( \Lambda^T y \geq 0, b^T y < 0 \) has a solution, where \( y \) is a vector.

**Proof.** If the system \( \Lambda^T y \geq 0, b^T y < 0 \) has a solution, then there exists \( \bar{y} \) such that \( \Lambda^T \bar{y} \geq 0, b^T \bar{y} < 0 \). Assume that there exists \( \bar{x} \) such that

\[
\Lambda \bar{x} = b,
\]

and \( \bar{x} \geq 0 \). Hence, \( \bar{x}^T \Lambda^T \bar{y} = b^T \bar{y} \). Since \( \bar{x} \geq 0 \) and \( \Lambda^T \bar{y} \geq 0, b^T \bar{y} < 0 \). This leads to a contradiction as before.

Let \( S = \{z \mid z = \Lambda x, x \geq 0\} \). \( S \) is a close convex set. If the system \( Ax = b, x \geq 0 \) has no solution, then there exists nonzero vector \( \bar{y} \) and \( \varepsilon > 0 \) such that \( \forall z \in S \),

\[
\bar{y}^T b + \varepsilon \leq \bar{y}^T z.
\]

Since \( \varepsilon > 0 \), from (19),

\[
\bar{y}^T b < \bar{y}^T z.
\]

Hence

\[
b^T \bar{y} < z^T \bar{y} = x^T \Lambda^T \bar{y}.
\]

Since \( b^T \bar{y} \) is fixed and \( x \geq 0 \) is arbitrary, \( b^T \bar{y} < 0, \Lambda^T \bar{y} \geq 0; i.e., the system \( \Lambda^T y \geq 0, b^T y < 0 \) has a solution. \( \square \)

**Theorem 1.** If there exists a nonzero vector \( b \) such that the system \( Ax = b, x \geq 0 \) has no solution, then there exist cloning templates \( \Lambda, \bar{B}, \bar{D} \) defined in (5), (6) and (7), respectively, such that \( T \bar{B} \bar{D} \) is not empty.

**Proof.** Since there exists a nonzero vector \( b \) such that the system \( Ax = b, x \geq 0 \) has no solution, according to Lemma 1, there exists a vector \( y \in \mathbb{R}^8 \) such that \( \Lambda^T y \geq 0, b^T y < 0 \). From (14), \( \forall i \in N_0, \forall r \in \mathcal{J}_i^+, \forall q \in \mathcal{J}_i^- \),

\[
(Q(r)(i) - Q(q)(i), P(r)(i) - P(q)(i))y \geq 0.
\]

Choose

\[
((l_{A+B}^{(i-1)}), (l_D))T = y,
\]

then

\[
(Q(r)(i) - Q(q)(i))l_{A+B}^{(i-1)} + (P(r)(i) - P(q)(i))l_D \geq 0.
\]

Since \( r \in \mathcal{J}_i^+, q \in \mathcal{J}_i^- \), \( s_1(r) = 1, s_2(q) = -1 \). From (23),

\[
(Q(r)(i) - Q(q)(i))l_{A+B} + (P(r)(i) - P(q)(i))l_D \geq 2.
\]

Hence, \( \forall i \in N_0, \forall r \in \mathcal{J}_i^+, \forall q \in \mathcal{J}_i^- \),

\[
1 - Q(r)(i)l_{A+B} - P(r)(i)l_D \leq -1 - Q(q)(i)l_{A+B} - P(q)(i)l_D.
\]
From (13), \( \forall i \in \mathcal{N}_0, v_i^+ \leq v_i^- \). From (11) and (12), \( \Pi_{i=1}^n[v_i^+, v_i^-] \) is not empty. 

**Remark 1.** According to the proof of Theorem 1, if there exist cloning templates \( A, B, D \) such that (23) holds, then \( \Pi_{i=1}^n[v_i^+, v_i^-] \) is not empty. Hence, it is very important to work out inequality (23). \( A \) is a \( 18 \times \xi \) matrix, where \( \xi = \sum_{i \in \mathcal{N}_0} |J_i^+| \times |J_i^-| \). Hence, from (23), we only need to solve \( \xi \) inequalities. Since \( |J_i^+| + |J_i^-| = p, |J_i^+| \times |J_i^-| \leq (|J_i^+| + |J_i^-|)^2/4 = p^2/4 \). When \( |\mathcal{N}_0| = n \), we need to solve \( np^2/4 \) inequalities at most.

Specially, we have the following corollary on autoassociative memories.

**Corollary 1.** If \( s(t) = u(t) \), then there exist cloning templates \( A, B, D \) defined in (5), (6) and (7), respectively, such that \( \Pi_{i=1}^n[v_i^+, v_i^-] \) is not empty.

**Proof.** Since \( s(t) = u(t) \),

\[
Q_r(i) = (P^{(r)}(i), Q^{(q)}(i) = P^{(q)}(i)).
\]

From \( r \in \mathcal{J}_i^+ \) and \( q \in \mathcal{J}_i^- \), \( s_i^{(r)} = 1, s_i^{(q)} = -1 \). Choose \( a_0.0 + b_0.0 + d_0.0 \) enough large, then there exist \( A, B, D \) such that (23) holds. According Remark 1, \( \Pi_{i=1}^n[v_i^+, v_i^-] \) is not empty.

**Remark 2.** The design procedure herein, based on duality in Lemma 1, results in a set of inequalities instead of a set of equations solved using matrix pseudoinverse in some existing approaches (e.g., [15], [16]). Since the solution space of inequalities is generally bigger than that of equalities, the new design procedure is easier to be carried out and the performance of designed associative memories can be made robust by choosing appropriate solutions of inequalities (i.e., parameters of designed associative memories).

**Proposition.** Let \((Q^{(r)}(i) - Q^{(q)}(i))l^{(1)}_{A+B}\) be. If \( l^{(1)}_{A+B} \) is known, then \( l_B \), which satisfies the inequality (23), can be obtained by computing the limit of a trajectory of the following differential equation.

\[
\frac{dz(t)}{dt} = -(P^{(r)}(i) - P^{(q)}(i))^T[(P^{(r)}(i) - P^{(q)}(i))(z(t)] + b - [(P^{(r)}(i) - P^{(q)}(i))(z(t) + b)],
\]

\[\text{(25)}\]

**Proof.** If there exists a vector \( z^* \) such that \((P^{(r)}(i) - P^{(q)}(i))z^* + b \geq 0\), then \( z^* \) is an equilibrium point of (25). Let \( V(z(t)) = \|z(t) - z^*\|^2 \). Hence,

\[
\frac{dV(z(t))}{dt} = -2(z(t) - z^*)^T(P^{(r)}(i) - P^{(q)}(i))T
\]

\[
\cdot [(P^{(r)}(i) - P^{(q)}(i))(z(t) + b) - [(P^{(r)}(i) - P^{(q)}(i))(z(t) + b)]
\]

\[
= -2[(P^{(r)}(i) - P^{(q)}(i))(z(t) + b) - [(P^{(r)}(i) - P^{(q)}(i))(z(t) + b)]
\]

\[
\cdot [(P^{(r)}(i) - P^{(q)}(i))(z(t) + b) - [(P^{(r)}(i) - P^{(q)}(i))(z(t) + b)]
\]

\[
< 0, \text{ otherwise, i.e., } dV(z(t))/dt_{(25)} < 0 \text{ except at the equilibrium points.}
\]

Thus, any trajectory of (25) can converge to an equilibrium point of (25).

**Theorem 2.** If there exists a nonzero vector \( c \) such that the system \( \Lambda(u)x = c, x \geq 0 \) has no solution and there exists a vector \( y_0 \) such that the system \( \Lambda(u)^T x = (\Lambda(u))^T y_0 \) has a solution, then there exist cloning templates \( A, B, D \), such that \( \Pi_{i=1}^n[v_i^+, v_i^-] \) is not empty.

**Proof.** Since there exists a nonzero vector \( c \) such that the system \( \Lambda(u)x = c, x \geq 0 \) has no solution, according to Lemma 1, there exists a vector \( y \in \mathbb{R}^q \) such that \( (\Lambda(u))^T y \geq 0, c^Ty < 0 \). From (15), \( \forall i \in \mathcal{N}_0, \forall r \in \mathcal{J}_i^+, \forall q \in \mathcal{J}_i^- \),

\[
(P^{(r)}(i) - P^{(q)}(i))y \geq 0.
\]

Choose \( l_{A+B}^{(-1)} = y_s, l_D = y - x, \)

\[\text{(26)}\]

where \( x \) is a solution of the system \( (\Lambda(u))^T x = (\Lambda(s))^T y_s \); i.e.,

\[
(\Lambda(u))^T x = (\Lambda(s))^T y_s.
\]

Then

\[
(Q^{(r)}(i) - Q^{(q)}(i))l_{A+B}^{(-1)} + (P^{(r)}(i) - P^{(q)}(i))l_D \geq 0.
\]

(29)

Similar to the proof of Theorem 1, \( \Pi_{i=1}^n[v_i^+, v_i^-] \) is not empty.

**C. Range of the Bias Vector**

**Theorem 3.** If there exists a nonzero vector \( b \) such that the system \( Ax = b, x \geq 0 \) has no solution, then there exists a region \( \Pi_{i=1}^n[v_i^+, v_i^-] \), such that for any bias vector \( v \in \Pi_{i=1}^n[v_i^+, v_i^-] \), any \( u^{(f)} (f \in \{1, 2, \cdots, p\}) \) and some initial conditions, the output vector of the designed DCNN with cloning templates \( A, B, D \) converge to \( s^{(f)} \), where \( v_i^+ \) and \( v_i^- (i \in \{1, 2, \cdots, n\}) \) are defined in (11), (12) and (13), respectively.

**Proof.** \( \forall f \in \{1, 2, \cdots, p\} \), choose

\[
\tilde{x}^{(f)}_i = Q^{(f)}(i)l_{A+B} + P^{(f)}(i)l_D + v_i.
\]

(30)

According to Theorem 1, \( \Pi_{i=1}^n[v_i^+, v_i^-] \) is not empty. \( \forall v \in \Pi_{i=1}^n[v_i^+, v_i^-] \), if \( s_i^{(f)} = 1 \), then from (11) and (13),

\[v_i \geq 1 - Q^{(f)}(i)l_{A+B} + P^{(f)}(i)l_D.
\]

(31)

Hence, when \( s_i^{(f)} = 1 \), from (30) and (31),

\[
\tilde{x}^{(f)}_i \geq 1, f(\tilde{x}_i^{(f)}) = s_i^{(f)}.
\]

(32)

Similarly, if \( s_i^{(f)} = -1 \), from then (12) and (13)

\[v_i \leq -1 - Q^{(f)}(i)l_{A+B} - P^{(f)}(i)l_D.
\]

(33)

Hence, when \( s_i^{(f)} = -1 \), from (30) and (33),

\[
\tilde{x}^{(f)}_i \leq -1, f(\tilde{x}_i^{(f)}) = s_i^{(f)}.
\]

(34)

From (9), (10), (32) and (34),

\[
\tilde{x}^{(f)} = Ag(\tilde{x}^{(f)}) +Bg(\tilde{x}^{(f)}) + Du^{(f)} + v,
\]

(35)
where, $\bar{x}(t) = (\bar{x}_1(t), \ldots, \bar{x}_n(t))^T$. From (3), $\bar{x}(t)$ is an equilibrium point of (1).

Let $\Omega(t) = \Pi_{i=1}^L L_i^{(t)}$, where

$$L_i^{(t)} = \begin{cases} [1, +\infty), & s_i^{(t)} = 1, \\ (-\infty, -1], & s_i^{(t)} = -1, \end{cases}$$

From (32) and (34), $\bar{x}(t) \in \Omega(t)$, $\forall r \in [t_0 - \tau, t_0]$, if $\phi(r) \in \Omega(t)$, then form (3) and (35),

$$\frac{dx(t)}{dt} = -x(t) + \bar{x}(t),$$

(36) is obviously globally exponentially stable. Hence, for any initial condition with $\phi(r) \in \Omega(t)$, $r \in [t_0 - \tau, t_0]$, the state of the neural network converges to $s^{(t)}$.

Remark 3. According to Theorem 3, the designed DCNN may be locally exponentially stable only. Hence, even if $u^{(t)}$ is input to the network, the state vector of the DCNN may not converge to $s^{(t)}$ for some initial conditions. In the following, some sufficient conditions are shown to ensure the state of the neural network converge to $s^{(t)}$ with $u^{(t)}$ being the input to the network.

Theorem 4. If there exist cloning templates $\hat{A}$ and $\hat{B}$ such that

$$\sum_{i=1}^{n} \sum_{j=1}^{n} \left( |\hat{a}_{ij}| + |\hat{b}_{ij}| \right) < 1,$$  \hspace{1cm} (37)

the system $(\Lambda(u))^{T} x = (\Lambda(s))^{T} l(t-1)$ has a solution, and there exists a nonzero vector $c$ such that the system $(\Lambda(u)) x = c, x \geq 0$ has no solution, then there exists cloning template $\hat{D}$ for any bias vector $v \in \Pi_{i=1}^{n}[v_i^+, v_i^-], (s^{(1)}, u^{(1)}), (s^{(2)}, u^{(2)}), \ldots, (s^{(p)}, u^{(p)})$ are $p$ pairwise associatively memorized vectors in the designed DCNN with cloning templates $\hat{A}, \hat{B}$ and $\hat{D}$.

Proof. \forall $l \in \{1, 2, \ldots, p\}$, choose

$$x_i^{(t)} = Q^{(t)}(l) i \in \hat{A} + \hat{B} + p(l) i \in \hat{D} + v.$$  \hspace{1cm} (38)

According to Theorem 2, $\Pi_{i=1}^{n}[v_i^+, v_i^-]$ is not empty. Similar to the proof of Theorem 3, $x(t) = (x_1(t), \ldots, x_n(t))^T$ is an equilibrium point of (3).

Since (37) holds, there exists $\theta > 0$ such that

$$1 - \sum_{i=1}^{n} \sum_{j=1}^{n} \left( |\hat{a}_{ij}| + |\hat{b}_{ij}| \exp(\theta \tau) \right) = 0.$$  \hspace{1cm} (39)

Let $y(t) = x(t) - \bar{x}(t)$. Then from (3),

$$\frac{dy(t)}{dt} = -y(t) + Ah(y(t)) + Bh(y(t) - \tau(t)),$$  \hspace{1cm} (40)

where $h(y(t)) = (h(y_1(t)), \ldots, h(y_n(t)))^T, h(y_i(t)) = f(y_i(t) + \bar{x}_i(t)) - f(\bar{x}_i(t)), i = 1, 2, \ldots, n$. From (2), $|h(y_i(t))| \leq |y_i(t)|$. By using the reduction to absurdity, the solution of (40) with initial condition $\phi(r), r \in [t_0 - \tau, t_0]$ satisfies for $i = 1, 2, \ldots, n, t \geq t_0$,

$$|y(t)| \leq ||\phi||_{l_0} \exp(-\theta(t - t_0)),$$

i.e., DCNN (3) has a globally exponentially stable equilibrium point. Hence, a pairwise vector $(s^{(t)}, u^{(t)})$ is associatively memorized by using DCNN (3). \hspace{1cm} $\square$

IV. DESIGN PROCEDURE WITH SPACE-INVARIANT DCNN

Step 1. Denote desired patterns by using vectors in $\{-1, 1\}^n$; i.e., determine $p$ pairwise vectors to be memorized $(s^{(1)}, u^{(1)}), (s^{(2)}, u^{(2)}), \ldots, (s^{(p)}, u^{(p)})$, where $\forall \ell \in \{1, 2, \ldots, p\}, s^{(\ell)}, u^{(\ell)} \in \{-1, 1\}^n$. $p$ is the number of the patterns and $n$ is the number of neurons of the DCNN.

Step 2. Compute $N_0, J_0^+, J_0^-$ according to the definitions on $N_0, J_0^+, J_0^-$ in Section III.

Step 3. Verify the condition of Theorem 4 or Corollary 6. If the conditions hold, then choose cloning templates $\hat{A}, \hat{B}$ and $\hat{D}$ such that (23) holds. Compute the connection weights $A, B, D$.

Step 4. Compute $\hat{I}_i^+$ and $\hat{I}_i^-$ ($i \in \{1, 2, \ldots, n\}$) according to (11), (12) and (13).

Step 5. Choose the bias vector $v \in \Pi_{i=1}^{n}[v_i^+, v_i^-]$, for $i \in \{1, 2, \ldots, n\}$.

Step 6. Synthesize the DCNN with the connection weights $A, B, D$ and bias vector $v$.

According to Theorem 4, $(s^{(1)}, u^{(1)}), (s^{(2)}, u^{(2)}), \ldots, (s^{(p)}, u^{(p)})$ can be $p$ pairwise heteroassociatively memorized vectors stored in the designed DCNN.

The allowable range of the inputs can be determined. For example, if $D\hat{u}(t)(i) + v_i \in [v_i^+, v_i^-]$, then $s^{(t)}$ is able to be recalled by the noisy input patterns $\hat{u}(t), \ell \in \{1, 2, \ldots, p\}$, where $D\hat{u}(t)(i)$ is the $i$-th element of the vector $D\hat{u}(t)$.

Hence, the designed associative memories are robust to parameter perturbation in pattern retrieval. In contrast, a set of equations must be solved by using matrix pseudoinverse in [15] and [16]. However, some equations may have no solution. In addition, the vector $v$ is fixed in advance in [15] and [16] (in this paper, the bias vector $v$ can be chosen in $\Pi_{i=1}^{n}[v_i^+, v_i^-]$). That may also result in no solution for the equations with fixed $v$.

V. SIMULATION RESULTS

In this section, we give two examples to illustrate the results.

Example 1. Let’s design a DCNN to behave as a heteroassociative memory. The input and output patterns are represented by two pairs of 6 × 6-pixel images (black pixel = -1; white pixel = 1), as shown in Fig. 1(a)-(b).

According to the definition on $N_0$ in Section III, $N_0 = \{3, 4, 5, 10, 15, 16, 17, 20, 26, 29, 30, 35\}, |N_0| = 12, |J_i^+| = |J_i^-| = 1, \xi = \sum_{i \in N_0} |J_i^+| \times |J_i^-| = 12 < 18.$
Choose the cloning templates

\[
\hat{A}_1 + \hat{B}_1 = \begin{bmatrix}
0.1 & 0.1 & 0.1 \\
0.1 & 0.1 & 0.1 \\
0.1 & 0.1 & 0.1 
\end{bmatrix},
\]

\[
\hat{D}_1 = \begin{bmatrix}
3.5319 & -4.4969 & 0.7757 \\
-2.8479 & -3.3903 & 0.6241 \\
-0.6577 & -2.1204 & 2.2408 
\end{bmatrix}.
\]

Then the conditions of Theorem 4 hold. Using (11), (12) and (13), for \(i \in \{1,2,\ldots,36\} \), compute \(\mathcal{I}_i^+\) and \(\mathcal{I}_i^-\). Let 
\[
v^- = (v_1, \ldots, v_{36})^T, \quad v^+ = (v_1^+, \ldots, v_{36}^+)^T.
\]

\(v^- = (+\infty, +\infty, 3.8768, 4.0768, 3.8360, +\infty, +\infty, +\infty, +\infty, +\infty, +\infty, +\infty, +\infty, +\infty, +\infty, +\infty, +\infty, +\infty, +\infty, +\infty, +\infty, +\infty, +\infty, +\infty, +\infty, +\infty, +\infty, +\infty, +\infty, +\infty, +\infty, +\infty, +\infty, 5.0034, 8.4032)^T\).

Choose \(v_5 = v_{29} = 2, v_{17} = 5.2, v_{30} = 9.3\). For \(i \in \{3,4,10,15,16,20,26,35\}\), choose \(v_i = 0\). For \(i \in \{1,2,6,7,8,9,12,13,14,18,19,21,22,24,25,31,32,33,34,36\}\), choose \(v_i = 18\). For \(i \in \{11,23,27,28\}\), choose \(v_i = -10\). Hence, \(v = (v_1, \ldots, v_{36})^T \in \Pi_{i=1}^{36} [v_i^+, v_i^-]\). Synthesize the DCNN with the cloning templates \(\hat{A}, \hat{B}, \hat{D}\), and bias vector \(v\). Then, according to Theorem 4, for any bounded time-delays \(\tau_{ij}(t)\), \((s^{(1)}(u^{(1)}), s^{(2)}(u^{(2)}))\) are 2 pairwise heteroassociatively memorized vectors of the designed DCNN.

Fault tolerance is a very important issue in associative memories. To demonstrate the fault tolerance capability of the designed associative memories, this section reports the statistics of successful retrievals of designed associative memories of various sizes and capacities with distorted input probes.

Simulations show that the designed DCNN is able to recall the output patterns shown in Fig. 1(b) starting from any random initial conditions. When input pattern is \(0.5u^{(2)}\), simulation results with 6 random initial values and input pattern being \(0.5u^{(2)}\) are depicted in Figure 2.

**Example 2**. Let’s design a DCNN to behave as an autoassociative memory. The input and output patterns are represented by 12 4×3-pixel images (black pixel = −1; white pixel = 1), as shown in Fig. 3.

According to the definitions on \(\mathcal{N}_0\) and \(\mathcal{N}_{-1}\) in Section III, \(\mathcal{N}_{-1} = \{1\}, \mathcal{N}_0 = \{2,3,\ldots,12\}, |\mathcal{N}_0| = 11\),

\[
\mathcal{J}_i^+ = \{1,\ldots,i-1, i+1,\ldots,12\}, \quad 1 < i < 12;
\]

\[
\mathcal{J}_i^- = \{i\}.
\]

\[
|\mathcal{J}_i^+| = 11, \quad |\mathcal{J}_i^-| = 1 \quad \text{for} \quad i = 2, 3, \ldots, 12, \quad \xi = \sum_{i \in \mathcal{N}_0} |\mathcal{J}_i^+| \times |\mathcal{J}_i^-| = 11 \times 11 = 121.
\]
Then the conditions of Theorem 4 hold. Using (11),

\[
\dot{v} = -v + (v_T - v) + Bv - u.
\]

on space-invariant cloning templates by solving a set of

output connection weight matrices can be obtained based

by feeding probes via external inputs rather than initial

synthesized with assured global exponential stability. In the

enables heteroassociative or autoassociative memories to be

sociative memories based on CNNs is presented, which

VI. CONCLUDING REMARKS

In this paper, a design procedure for synthesizing asso-

ciative memories based on CNNs is presented, which

enables heteroassociative or autoassociative memories to be

synthesized with assured global exponential stability. In the

designed associative memories, stored patterns are retrieved

by feeding probes via external inputs rather than initial

states. In the new design procedure, the designed input and

output connection weight matrices can be obtained based

on space-invariant cloning templates by solving a set of

inequalities. In addition, the allowable range of the inputs

is determined. Hence, the designed associative memories are

robust to parameter perturbation in pattern retrieval. Hence,

the designed parameters are few even though the dimension

of desired patterns may be very high. Simulation results

illustrate the usage of the proposed approach.

REFERENCES


properties of neural networks: new learning mechanisms,” Physical


with lower block triangular interconnecting structure,” Trans. 


associative memories with applications to cellular neural networks,” 


networks for associative memories with learning and forgetting capabil-


1995.


based on recurrent neural networks with linear saturation activation 

functions and time-varying delays,” Neural Computation, vol. 19, 


associative memories based on a class of discrete-time recurrent neural 


networks with application to linear and quadratic programming prob-


a general class of recurrent neural networks with time-varying delays,” 


[21] S.-C. Fang and S. Puthenpura, Linear Optimization and Extensions: 