An Improvement of the Design Method of Cellular Neural Networks Based on Generalized Eigenvalue Minimization

Author(s): Ryoma Bise and Norikazu Takahashi

Journal: IEEE Transactions on Circuits and Systems I
Volume: 50
Number: 12
Pages: 1569–1574
Month: December
Year: 2003
Published Version: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1257462

An Improvement of the Design Method of Cellular Neural Networks based on Generalized Eigenvalue Minimization

Ryoma Bise , Norikazu Takahashi, and Tetsuo Nishi

***Abstract*— Realization of associative memories by cellular neural networks (CNNs) with binary output is studied. Concerning this problem, a CNN design method based upon generalized eigenvalue minimization (GEVM) has recently been proposed. In this paper, a new CNN design method which is based on the GEVM-based method will be presented. We first give some analytical results related to the basin of attraction of a memory vector. We then derive the design method by combining those analytical results and the GEVM-based method. We finally show through computer simulations that the proposed method can achieve higher recall probability than the original GEVM-based method.**

***Index Terms*— Cellular neural networks, associative memory, basin of attraction, generalized eigenvalue minimization.**

I. Introduction

A cellular neural network (CNN) is a nonlinear analog circuit consisting of a number of locally coupled signal processing elements called cells. Since the first paper of Chua and Yang [1] was published in 1988, CNNs have found many applications mainly in the field of image processing. In each application of CNNs, it is very important to find optimum values of the network parameters so that a CNN performs a desired task. So far, there have been many attempts to construct systematic ways of designing CNNs with space-invariant couplings for image processing tasks [2], [3], [4], [5].

In this paper, as a fundamental design problem for CNNs, we consider the realization of associative memories by means of CNNs with space-varying couplings. As is well known, this problem has been vigorously studied for fully coupled neural networks such as the Hopfield model from the early ninety-eighties, and various design methods have been proposed so far [6], [7], [8], [9], [10], [11]. However, since each cell in a CNN is connected only with its neighboring cells, these methods cannot be applied directly to CNNs. In order to make use of CNNs for associative memories, it is thus required to develop design methods suitable for their structural characteristics. Liu and Michel [12] have proposed a design method for sparsely interconnected neural networks and applied it to CNNs. Their method is a modification of the eigenstructure method [10] which is well known as an effective design technique for fully connected neural networks. Seiler *et al.* [13] developed an optimization-based method for realizing prescribed stable output patterns and unstable output patterns of a CNN. The method proposed by Grassi [14], [15] is distinct from others because it makes use of CNNs possessing a unique equilibrium point which is globally asymptotically stable and input patterns are fed into such CNNs as bias vectors.

Park *et al.* [16] have recently proposed a method to design CNNs based on the generalized eigenvalue minimization (GEVM) [17], [18], [19]. Throughout this paper, this method will be referred to as Park's method for simplicity. In the synthesis procedure of Park's method, for each step the feasibility of a set of linear inequalities is first checked. If it is feasible, a GEVM problem is solved to find the network parameters which guarantee both that the prototype vectors are stored as memory vectors and that the basin of attraction of each prototype vector is maximized in a certain sense. Otherwise, a linear matrix inequality (LMI) problem is solved to find the network parameters which guarantee just that the prototype vectors are stored as memory vectors. Since both GEVM problems and LMI problems are solved efficiently by using computer software such as MATLAB [20], the network parameters can be easily obtained. The results of computer simulations carried out by Park *et al.* [16] have shown that Park's method is superior in the average recall probability to the modified eigenstructure method proposed by Liu and Michel. Therefore, Park's method can be regarded as one of the most effective CNN design techniques for associative memories. We note here that LMIs also play important roles in the stability analysis of recurrent neural networks; for example, LMI conditions are given by Suykens *et al.* [21], [22] in relation to the global asymptotic stability of multilayer recurrent neural networks and the basins of attraction of equilibrium points.

In this paper, we propose a new CNN design procedure based on Park's method. We first give some analytical results related to the basin of attraction of a memory vector. One of them is a generalization of a theorem given by Park *et al.* [16], and the others are firstly obtained in the present paper. We then develop the CNN design procedure by modifying a certain part of Park's method by using those analytical results. We finally show through computer simulations that the proposed method can achieve higher recall probability than the original GEVM-based method.

II. Problem Formulation

Let us consider CNNs described by the following differential equations:

$$\frac{\mathrm{d}x_i}{\mathrm{d}t} = -x_i + \sum_{j \in \bar{N}_i} A_{ij} y_j + I_i, \quad i = 1, 2, \dots, n \tag{1}$$

where x_i is the state of the i-th cell, y_i the output of the i-th cell determined by x_i through

$$y_i = f(x_i) \triangleq \frac{1}{2}(|x_i + 1| - |x_i - 1|), \tag{2}$$

A_{ij} the coupling coefficient from the j-th cell to the i-th cell, I_i the bias of the i-th cell, and $\bar{N}_i$ the set of indices of the cells belonging to the neighborhood of the i-th cell. Although it is

A previous version of this paper appeared in the Proceedings of the 7th IEEE International Workshop on Cellular Neural Networks and their Applications, pp.515–522, 2002. This research was partially supported by Japan Society for the Promotion of Science, Grant-in-Aid for Encouragement of Young Scientists, 15760268, 2003 and the 21st Century COE Program 'Reconstruction of Social Infrastructure Related to Information Science and Electrical Engineering'.

R. Bise is with the Dai Nippon Printing Co. Ltd., Tokyo, Japan (email: Bise-R@mail.dnp.co.jp)

N. Takahashi and T. Nishi are with the Department of Computer Science and Communication Engineering, Kyushu University, Fukuoka, Japan (email: norikazu@csce.kyushu-u.ac.jp; nishi@csce.kyushu-u.ac.jp)

often assumed in CNN literature that coupling coefficients between cells are space-invariant [1], we will not make this assumption in this paper. Note that the sets $\bar{N}_i,\ i = 1, 2, \ldots, n$ are uniquely determined by the method of numbering cells and the radius of neighborhood denoted by r. In the following, the neighborhood of the i-th cell excluding itself is represented by N_i, that is, $N_i \triangleq \bar{N}_i \setminus \{i\}$ for convenience. Also, the set of n dimensional real vectors, the set of $n \times n$ real matrices, and the set of n dimensional binary vectors, i.e. the vectors whose elements are $+1$ or -1 are denoted by $\mathbb{R}^n$, $\mathbb{R}^{n\times n}$ and $\mathbb{B}^n$, respectively. By introducing the state vector $\boldsymbol{x} = [x_1, x_2, \ldots, x_n]^T$, the output vector $\boldsymbol{y} = [y_1, y_2, \ldots, y_n]^T$, the connection matrix $\boldsymbol{A} = [A_{ij}] \in \mathbb{R}^{n\times n}$, the bias vector $\boldsymbol{I} = [I_1, I_2, \ldots, I_n]^T$, the piecewise linear mapping $\boldsymbol{f}(\boldsymbol{x}) \triangleq [f(x_1), f(x_2), \ldots, f(x_n)]^T$, and the set of matrices $\mathcal{M}\left(\bar{N}_1, \bar{N}_2, \ldots, \bar{N}_n\right)$ defined by

$$\mathcal{M}\left(\bar{N}_1, \bar{N}_2, \ldots, \bar{N}_n\right) \triangleq \left\{\boldsymbol{S} = [S_{ij}] \in \mathbb{R}^{n\times n} \mid S_{ij} = 0 \ \text{ if } \ j \notin \bar{N}_i\right\},$$

we can rewrite (1) and (2) in vector form as follows:

$$\frac{\mathrm{d}\boldsymbol{x}}{\mathrm{d}t} = -\boldsymbol{x} + \boldsymbol{A}\boldsymbol{y} + \boldsymbol{I} \tag{3}$$

$$\boldsymbol{y} = \boldsymbol{f}(\boldsymbol{x}) \tag{4}$$

where the connection matrix $\boldsymbol{A}$ belongs to $\mathcal{M}\left(\bar{N}_1, \bar{N}_2, \ldots, \bar{N}_n\right)$. A vector $\boldsymbol{y}^e$ is referred to as a memory vector of the CNN described by (3) and (4) if the CNN has an asymptotically stable equilibrium point $\boldsymbol{x}^e$ such that $\boldsymbol{y}^e = \boldsymbol{f}(\boldsymbol{x}^e)$. The set of initial states $\boldsymbol{x}(0)$ such that $\lim_{t\to\infty} \boldsymbol{x}(t) = \boldsymbol{x}^e$ is called the basin of attraction of the memory vector $\boldsymbol{y}^e$. As is well known, if every diagonal element of $\boldsymbol{A}$ is greater than or equal to unity, then asymptotically stable equilibrium points can exist only in the total saturation region which is defined by $\{\boldsymbol{x} \in \mathbb{R}^n \mid |x_i| \geq 1,\ \forall i\}$ [1], [23]. Thus, in this case, all memory vectors are binary.

Based on the design problem given by Michel and Liu [11] for fully coupled neural networks, we formulate the CNN design problem for associative memories as follows:

CNN Design Problem: For given prototype vectors $\boldsymbol{\alpha}^1, \boldsymbol{\alpha}^2, \ldots, \boldsymbol{\alpha}^m \in \{1, -1\}^n$ and the sets $\bar{N}_1, \bar{N}_2, \ldots, \bar{N}_n$, find the connection matrix $\boldsymbol{A} \in \mathcal{M}\left(\bar{N}_1, \bar{N}_2, \ldots, \bar{N}_n\right)$ and the bias vector $\boldsymbol{I}$ such that the synthesized CNN has the following properties.

1) All prototype vectors $\boldsymbol{\alpha}^1, \boldsymbol{\alpha}^2, \ldots, \boldsymbol{\alpha}^m$ are memory vectors.
2) The total number of spurious memory vectors, that is, the memory vectors of the CNN not contained in $\{\boldsymbol{\alpha}^1, \boldsymbol{\alpha}^2, \ldots, \boldsymbol{\alpha}^m\}$, is as small as possible.
3) The basin of attraction of each prototype vector is as large as possible.
4) The CNN has no oscillatory solution.

Since we will not assume that the connection matrix $\boldsymbol{A}$ is symmetric or satisfies other stability conditions obtained so far [24], [25], [26], the fourth property is not satisfied in general. We will thus focus our attention only on the first three properties. Note here that the second and third properties are closely related because the basins of attraction of the prototype vectors become larger, as the total number of spurious memory vectors becomes smaller, and vice versa. Note also that there will be various strategies for designing CNNs depending on what kind of qualitative criterion for the sizes of the basins of attraction is considered. For example, Park's method designs a CNN so that the smallest basin of attraction is maximized in a certain sense. In this case, the design problem is formulated as a kind of minmax problem.

III. Analysis

In this section, we will present some theorems concerning the basin of attraction of a memory vector of a CNN which play important roles in the CNN design method given in the next section.

Theorem 1: Suppose n sets $\bar{N}_1, \bar{N}_2, \ldots, \bar{N}_n$ and a binary vector $\boldsymbol{\alpha}^* = [\alpha_1^*, \alpha_2^*, \ldots, \alpha_n^*]^T \in \mathbb{B}^n$ are given. If the connection matrix $\boldsymbol{A} = [A_{ij}] \in \mathcal{M}\left(\bar{N}_1, \bar{N}_2, \ldots, \bar{N}_n\right)$ and the bias vector $\boldsymbol{I}$ satisfy

$$\alpha_i^*\left(\sum_{j\in N_i} A_{ij}\alpha_j^* + I_i\right) > \kappa_i \max_{j\in N_i}|A_{ij}| + (A_{ii} - 1) \tag{5}$$

with $A_{ii} \geq 1$ and $\kappa_i \geq 0$, then any vector $\boldsymbol{\beta} = [\beta_1, \beta_2, \ldots, \beta_n]^T \in \mathbb{R}^n$ such that $f(\beta_i) \neq \alpha_i^*$ and $\sum_{j\in N_i}|f(\beta_j) - \alpha_j^*| \leq \kappa_i$ has the following properties.

1) The vector $\boldsymbol{\alpha} = \boldsymbol{f}(\boldsymbol{\beta})$ is not a memory vector.
2) If $\boldsymbol{x}(0) = \boldsymbol{\beta}$ then $x_i(t)$ moves toward α_i^* at $t = 0$.

Proof: It follows from the state equation of a CNN that

$$\begin{aligned}\alpha_i^* \cdot \left.\frac{\mathrm{d}x_i}{\mathrm{d}t}\right|_{\boldsymbol{x}=\boldsymbol{\beta}} &= \alpha_i^*\left(-\beta_i + \sum_{j\in\bar{N}_i} A_{ij}f(\beta_j) + I_i\right)\\ &= \alpha_i^*\left(\sum_{j\in\bar{N}_i} A_{ij}f(\beta_j) + I_i\right) - \alpha_i^*\beta_i.\end{aligned} \tag{6}$$

By applying the conditions to the first term on the right-hand side, we have

$$\begin{aligned}&\alpha_i^*\left(\sum_{j\in\bar{N}_i} A_{ij}f(\beta_j) + I_i\right)\\ =\ & A_{ii}\alpha_i^* f(\beta_i) + \alpha_i^*\left(\sum_{j\in N_i} A_{ij}f(\beta_j) + I_i\right)\\ =\ & A_{ii}\alpha_i^* f(\beta_i) + \alpha_i^*\Bigg(\sum_{j\in N_i} A_{ij}\alpha_j^* + I_i\\ &\quad + \sum_{j\in N_i} A_{ij}(f(\beta_j) - \alpha_j^*)\Bigg)\\ \geq\ & A_{ii}\alpha_i^* f(\beta_i) + \alpha_i^*\left(\sum_{j\in N_i} A_{ij}\alpha_j^* + I_i\right)\\ &\quad - \left|\sum_{j\in N_i} A_{ij}(f(\beta_j) - \alpha_j^*)\right|\end{aligned}$$

$$
\begin{aligned}
&\geq A_{ii}\alpha_i^* f(\beta_i) + \alpha_i^* \left(\sum_{j \in N_i} A_{ij}\alpha_j^* + I_i \right) - \kappa_i \max_{j \in N_i} |A_{ij}| \\
&= (1 + \alpha_i^* f(\beta_i)) A_{ii} - 1 + \alpha_i^* \left(\sum_{j \in N_i} A_{ij}\alpha_j^* + I_i \right) \\
&\quad -\kappa_i \max_{j \in N_i} |A_{ij}| - (A_{ii} - 1) \\
&> (1 + \alpha_i^* f(\beta_i)) A_{ii} - 1 \\
&\geq \alpha_i^* f(\beta_i). \qquad (7)
\end{aligned}
$$

From (6) and (7) the following inequality is obtained.

$$
\alpha_i^* \cdot \left. \frac{\mathrm{d}x_i}{\mathrm{d}t} \right|_{\boldsymbol{x}=\boldsymbol{\beta}} > \alpha_i^* f(\beta_i) - \alpha_i^* \beta_i \qquad (8)
$$

Since it is assumed that $f(\beta_i) \neq \alpha_i^*$, there are two possible cases where $|f(\beta_i)| < 1$ and $f(\beta_i) = -\alpha_i^*$. In the former case, the right-hand side of (8) vanishes because $f(\beta_i) = \beta_i$ holds. In the latter case, the right-hand side of (8) takes a nonnegative value because $\alpha_i^* f(\beta_i) = -1 \geq \alpha_i^* \beta_i$ holds. Therefore, (6) always takes a positive value. This means that $\boldsymbol{\beta}$ is not an equilibrium point and that if $\boldsymbol{x}(0) = \boldsymbol{\beta}$ then $x_i(t)$ tends to α_i^* at $t = 0$. ■

Although Theorem 1 is very similar to a theorem given by Park *et al.* [16], these two results are different at the following two points. One is that Theorem 1 contains the condition $A_{ii} \geq 1$ while in the theorem of Park *et al.* no condition is assumed on the value of A_{ii}. Since, as far as only binary output is concerned, it is common in designing CNNs to assume $A_{ii} \geq 1$, adding this condition will not lose generality. In fact, $A_{ii} \geq 1$ is assumed even in Park's method. The other is that $\boldsymbol{\beta}$ is assumed to be a real vector in Theorem 1 while $\boldsymbol{\beta}$ is restricted to a binary vector in the theorem of Park *et al.*. Because of this difference, Theorem 1 can be considered as an extension of the theorem of Park *et al.*.

Theorem 2: Suppose n sets $\bar{N}_1, \bar{N}_2, \ldots, \bar{N}_n$ and a binary vector $\boldsymbol{\alpha}^* = [\alpha_1^*, \alpha_2^*, \ldots, \alpha_n^*]^T \in \mathbb{B}^n$ are given. If the connection matrix $\boldsymbol{A} = [A_{ij}] \in \mathcal{M}\left(\bar{N}_1, \bar{N}_2, \ldots, \bar{N}_n\right)$ and the bias vector $\boldsymbol{I}$ satisfy

$$
\alpha_i^* \left(\sum_{j \in N_i} A_{ij}\alpha_j^* + I_i \right) > \kappa_i \max_{j \in \bar{N}_i} |A_{ij}| - (A_{ii} - 1) \qquad (9)
$$

with $\kappa_i \geq 0$, then any vector $\boldsymbol{\beta} \in \mathbb{R}^n$ such that $f(\beta_i) \neq \alpha_i^*$ and $\sum_{j \in \bar{N}_i} |f(\beta_j) - \alpha_j^*| \leq \kappa_i$ has the followings properties.

1) The vector $\boldsymbol{\alpha} = \boldsymbol{f}(\boldsymbol{\beta})$ is not a memory vector.
2) If $\boldsymbol{x}(0) = \boldsymbol{\beta}$ then $x_i(t)$ moves toward α_i^* at $t = 0$.

Proof: By applying the condition to the first term on the right-hand side of (6), we have

$$
\begin{aligned}
&\alpha_i^* \left(\sum_{j \in \bar{N}_i} A_{ij} f(\beta_j) + I_i \right) \\
&= \alpha_i^* \left(\sum_{j \in \bar{N}_i} A_{ij}\alpha_j^* + I_i + \sum_{j \in \bar{N}_i} A_{ij}(f(\beta_j) - \alpha_j^*) \right) \\
&\geq \alpha_i^* \left(\sum_{j \in \bar{N}_i} A_{ij}\alpha_j^* + I_i \right) - \kappa_i \max_{j \in \bar{N}_i} |A_{ij}| \\
&= \alpha_i^* \left(\sum_{j \in N_i} A_{ij}\alpha_j^* + I_i \right) - \kappa_i \max_{j \in \bar{N}_i} |A_{ij}| + (A_{ii} - 1) + 1 \\
&> 1 \\
&\geq \alpha_i^* f(\beta_i).
\end{aligned}
$$

The rest of the proof is same as Theorem 1. ■

The following theorem follows from Theorems 1 and 2.

Theorem 3: Let $\mathcal{I}_1$ and $\mathcal{I}_2$ be two sets such that $\mathcal{I}_1 \cup \mathcal{I}_2 = \{1, 2, \ldots, n\}$ and $\mathcal{I}_1 \cap \mathcal{I}_2 = \phi$. Suppose n sets $\bar{N}_1, \bar{N}_2, \ldots, \bar{N}_n$ and a binary vector $\boldsymbol{\alpha}^* = [\alpha_1^*, \alpha_2^*, \ldots, \alpha_n^*]^T \in \mathbb{B}^n$ are given. If the connection matrix $\boldsymbol{A} \in \mathcal{M}\left(\bar{N}_1, \bar{N}_2, \ldots, \bar{N}_n\right)$ and the bias vector $\boldsymbol{I}$ of a CNN satisfy (5) with $A_{ii} \geq 1$ and $\kappa_i \geq 0$ for all $i \in \mathcal{I}_1$, and satisfy (9) with $\kappa_i \geq 0$ for all $i \in \mathcal{I}_2$, then $\boldsymbol{\alpha}^*$ is a memory vector of the CNN and any vector $\boldsymbol{\beta} \in \mathbb{R}^n$ satisfying

$$
\left.
\begin{aligned}
\sum_{j \in N_i} |f(\beta_j) - \alpha_j^*| \leq \kappa_i, \ \forall i \in \mathcal{I}_1 \\
\sum_{j \in \bar{N}_i} |f(\beta_j) - \alpha_j^*| \leq \kappa_i, \ \forall i \in \mathcal{I}_2
\end{aligned}
\right\} \qquad (10)
$$

belongs to the basin of attraction of $\boldsymbol{\alpha}^*$.

Proof: We first prove that $\boldsymbol{\alpha}^*$ is a memory vector. Since (5) is satisfied with $A_{ii} \geq 1$ and $\kappa_i \geq 0$ for any $i \in \mathcal{I}_1$, we have

$$
\begin{aligned}
\alpha_i^* \left(\sum_{j \in \bar{N}_i} A_{ij}\alpha_j^* + I_i \right) &= A_{ii} + \alpha_i^* \left(\sum_{j \in N_i} A_{ij}\alpha_j^* + I_i \right) \\
&> A_{ii} + \kappa_i \max_{j \in N_i} |A_{ij}| + (A_{ii} - 1) \\
&\geq 2A_{ii} - 1 \\
&\geq 1
\end{aligned}
$$

for all $i \in \mathcal{I}_1$. Since (9) is satisfied with $\kappa_i \geq 0$ for any $i \in \mathcal{I}_2$, we also have

$$
\begin{aligned}
\alpha_i^* \left(\sum_{j \in \bar{N}_i} A_{ij}\alpha_j^* + I_i \right) &= A_{ii} + \alpha_i^* \left(\sum_{j \in N_i} A_{ij}\alpha_j^* + I_i \right) \\
&> A_{ii} + \kappa_i \max_{j \in \bar{N}_i} |A_{ij}| - (A_{ii} - 1) \\
&\geq 1
\end{aligned}
$$

for all $i \in \mathcal{I}_2$. It follows from the above two inequalities that

$$
\alpha_i^* \left(\sum_{j \in \bar{N}_i} A_{ij}\alpha_j^* + I_i \right) > 1, \quad i = 1, 2, \ldots, n
$$

which means that there exists an equilibrium point $\boldsymbol{x}^*$ such that $\boldsymbol{f}(\boldsymbol{x}^*) = \boldsymbol{\alpha}^*$. Since this equilibrium point lies inside the total saturation region, it is asymptotically stable. Therefore, $\boldsymbol{\alpha}^*$ is a memory vector of the CNN. Next, we will prove the second statement of the theorem. Let $\mathcal{R}$ be the set of vectors $\boldsymbol{\beta} \in \mathbb{R}^n$ satisfying (10). It follows from Theorems 1 and 2 that if $\boldsymbol{x}(t_0) \in \mathcal{R}$ then

$$
\frac{\mathrm{d}}{\mathrm{d}t} |f(x_i(t)) - \alpha_i^*| \leq 0, \quad i = 1, 2, \ldots, n \qquad (11)
$$

holds at $t = t_0$. The condition (11) holds with an equal sign if and only if $f(x_i(t_0)) = \alpha_i^*$. As a consequence, if $\boldsymbol{x}(0) \in \mathcal{R}$ then $\boldsymbol{x}(t) \in \mathcal{R}$ for all $t \geq 0$ and $\boldsymbol{f}(\boldsymbol{x}(t))$ converges to $\boldsymbol{\alpha}^*$ monotonically. ■

IV. Design

In the CNN design problem given in Section II, the most important thing is to store m prototype vectors $\boldsymbol{\alpha}^1, \boldsymbol{\alpha}^2, \ldots, \boldsymbol{\alpha}^m$ as memory vectors. This is achieved by choosing the connection matrix $\boldsymbol{A} \in \mathcal{M}(\bar{N}_1, \bar{N}_2, \ldots, \bar{N}_n)$ and the bias vector $\boldsymbol{I}$ such that the set of inequalities

$$\alpha_i^k \left(\sum_{j \in \bar{N}_i} A_{ij} \alpha_j^k + I_i \right) > 1, \quad k = 1, 2, \ldots, m \tag{12}$$

holds for $i = 1, 2, \ldots, n$. Note that (12) is feasible with $A_{ii} = 1 + \varepsilon$ $(\varepsilon > 0)$ for any set of the prototype vectors $\boldsymbol{\alpha}^1, \boldsymbol{\alpha}^2, \ldots, \boldsymbol{\alpha}^m$ because the left-hand side of (12) becomes $1 + \varepsilon$ if we put $A_{ij} = 0$, $\forall j \in N_i$ and $I_i = 0$. It is also important in the CNN design problem to guarantee that memory vectors are restricted to be binary. As we have mentioned before, this is achieved by choosing the values of the diagonal elements of $\boldsymbol{A}$ such that

$$A_{ii} \geq 1, \quad i = 1, 2, \ldots, n. \tag{13}$$

The main idea of Park's method is to try to make the basins of attraction of the prototype vectors as large as possible by making use of Theorem 1 while guaranteeing the above two requirements, i.e., (12) and (13). The total number of spurious memory vectors is expected to be reduced if the basins of attraction become large. Let us consider the set of inequalities

$$\alpha_i^k \left(\sum_{j \in N_i} A_{ij} \alpha_j^k + I_i \right) > \kappa_i \max_{j \in N_i} |A_{ij}| + (A_{ii} - 1), \qquad k = 1, 2, \ldots, m \tag{14}$$

which corresponds to (5) in Theorem 1. If κ_i can be maximized under the constraints (14), $\kappa_i \geq 0$ and $A_{ii} \geq 1$ for $i = 1, 2, \ldots, n$, then the basins of attraction of the prototype vectors are maximized in some sense. Note here that κ_i is maximized when $A_{ii} = 1$. Note also that (14) is feasible with $\kappa_i \geq 0$ and $A_{ii} = 1$ if and only if

$$\alpha_i^k \left(\sum_{j \in N_i} A_{ij} \alpha_j^k + I_i \right) > 0, \quad k = 1, 2, \ldots, m \tag{15}$$

is feasible. Furthermore it is apparent that (12) is feasible with $A_{ii} = 1$ if and only if (15) is feasible. Park's method thus first checks the feasibility of (15). If (15) is feasible, A_{ii} is set to 1 and the optimization problem:

$$\begin{aligned} &\text{Minimize} \quad -\kappa_i \\ &\text{Subject to} \quad -\kappa_i q_i + \alpha_i^k \left(\sum_{j \in N_i} A_{ij} \alpha_j^k + I_i \right) > 0, \quad k = 1, 2, \ldots, m \\ &\qquad |A_{ij}| < q_i, \quad \forall j \in N_i \\ &\qquad |I_i| < U \\ &\qquad L < q_i < U. \end{aligned} \tag{16}$$

is solved, where L and U $(L < U)$ are positive constants specified by users. The optimization problem (16) is in the form of the GEVM problem. If, on the other hand, (15) is not feasible, since (14) is not feasible with $\kappa_i \geq 0$ and $A_{ii} \geq 1$ in this case, A_{ii} is set to $1 + \varepsilon$ $(\varepsilon > 0)$ and the values of A_{ij}, $j \in N_i$ and I_i satisfying (12) with $A_{ii} = 1 + \varepsilon$ are found. One can easily see that this is an LMI problem.

Although computer simulation results [16] show that Park's method can achieve much higher average recall probability than the modified eigenstructure method [12], it is still insufficient because the basins of attraction of prototype vectors are not taken into account at all in the case where (15) is not feasible. In order to solve this problem, we make use of Theorem 2 in addition to Theorem 1.

Let us consider the set of inequalities

$$\alpha_i^k \left(\sum_{j \in N_i} A_{ij} \alpha_j^k + I_i \right) > \kappa_i \max_{j \in \bar{N}_i} |A_{ij}| - (A_{ii} - 1), \qquad k = 1, 2, \ldots, m \tag{17}$$

which corresponds to (9) in Theorem 2. If we set $A_{ii} = 1 + \varepsilon$ $(\varepsilon > 0)$, $A_{ij} = 0$, $\forall j \in N_i$, $I_i = 0$ and $\kappa_i = 0$ then the left-hand side of (17) becomes 0, while the right-hand side becomes $-\varepsilon$. This means that the set of inequalities (17) is feasible for any set of prototype vectors $\boldsymbol{\alpha}^1, \boldsymbol{\alpha}^2, \ldots, \boldsymbol{\alpha}^m$ under the conditions that $A_{ii} \geq 1$ and $\kappa_i \geq 0$. Therefore, even if (15) is not feasible, we can make the basins of attraction of the prototype vectors as large as possible by setting $A_{ii} = 1 + \varepsilon$ and solving the optimization problem

$$\begin{aligned} &\text{Minimize} \quad -\kappa_i \\ &\text{Subject to} \quad -\kappa_i q_i + \alpha_i^k \left(\sum_{j \in N_i} A_{ij} \alpha_j^k + I_i \right) > -\varepsilon, \quad k = 1, 2, \ldots, m \\ &\qquad |A_{ij}| < q_i, \quad \forall j \in N_i \\ &\qquad |I_i| < U \\ &\qquad L < q_i < U \end{aligned} \tag{18}$$

where L and U $(L < U)$ are positive constants specified by users. The optimization problem (18) is in the form of the GEVM problem.

From the above considerations, we derive the following CNN design method.

CNN Design Method: Given n sets $\bar{N}_1, \bar{N}_2, \ldots, \bar{N}_n \subseteq \{1, 2, \ldots, n\}$ and m prototype vectors $\boldsymbol{\alpha}^1, \boldsymbol{\alpha}^2, \ldots, \boldsymbol{\alpha}^m \in \mathbb{B}^n$, execute the following procedure for $i = 1, 2, \ldots, n$.

1) Check whether the set of inequalities (15) is feasible. If it is feasible go to Step 2), otherwise go to Step 3).
2) Set $A_{ii} = 1$, and find A_{ij}, $j \in N_i$ and I_i by solving the optimization problem (16).
3) Set $A_{ii} = 1 + \varepsilon$ $(\varepsilon > 0)$, and find A_{ij}, $j \in N_i$ and I_i by solving the optimization problem (18).

The proposed method tries to maximize κ_i in Theorem 2 even in the case where the set of inequalities (15) is not feasible, while Park's method just tries to satisfy the condition (12). It is thus expected from Theorem 3 that the proposed method

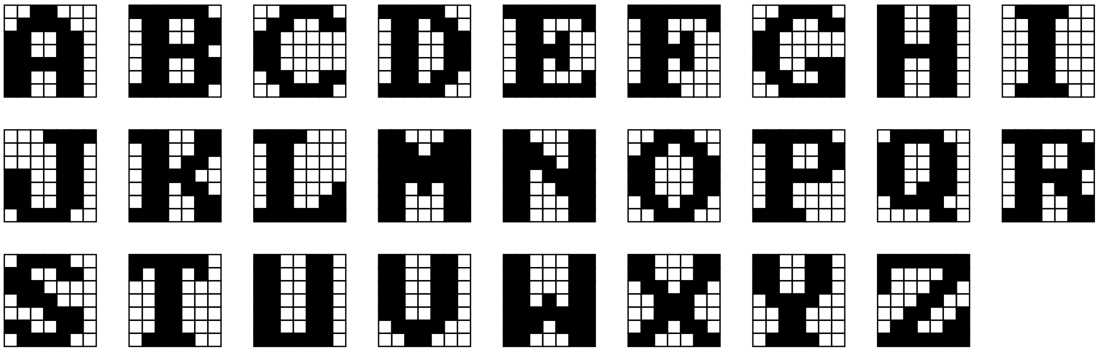

Fig. 1. Prototype vectors

TABLE I

COMPARISON OF THE AVERAGE RECALL PROBABILITY $P_{\mathrm{AV}}(d)$ BETWEEN PARK'S METHOD AND THE PROPOSED METHOD

radius of neighborhood	design method	Average recall probability $P_{\mathrm{av}}(d)$				
		$d=1$	$d=2$	$d=3$	$d=4$	$d=5$
$r=1$	Park's method	0.4623	0.2239	0.1109	0.0566	0.0288
	Proposed method	0.5479	0.2969	0.1623	0.0879	0.0455
$r=2$	Park's method	0.9082	0.8186	0.7257	0.6319	0.5369
	Proposed method	0.9584	0.8898	0.8046	0.7080	0.6026
$r=3$	Park's method	0.9812	0.9576	0.9347	0.9058	0.8651
	Proposed method	0.9937	0.9766	0.9585	0.9298	0.8891

can realize larger basins of attraction of the prototype vectors than Park's method. Moreover, since the optimization problems (16) and (18) are in the form of the GEVM problem, they can be efficiently solved numerically, as in Park's method, by using interior-point algorithms.

Let us now consider the computational complexity of the proposed method. First, the GEVM problems (16) and (18) can be solved in polynomial time in the size of the problem, which is determined by the number of prototype vectors and the radius of neighborhood, by using some efficient optimization techniques [18]. Second, the computation time of the proposed method is proportional to the number of cells as far as both the number of prototype vectors and the radius of neighborhood are fixed. From these facts, we can conclude that the proposed method can be solved in linear time in the number of cells, and in polynomial time in the number of prototype vectors and the radius of neighborhood. It is thus expected that the proposed method is applicable for CNNs with a large number of cells.

V. COMPUTER SIMULATIONS

In order to verify efficiency of the proposed method, we apply Park's method and the proposed method to the same set of prototype vectors, and compare the performance of synthesized CNNs in terms of the average recall probability [16] which is defined as follows: Let $\boldsymbol{\alpha}^k$ be any prototype vector of a CNN. The probability that the CNN converges to $\boldsymbol{\alpha}^k$ when the initial state $\boldsymbol{x}(0)$ is randomly chosen from the binary vectors such that the Hamming distance between $\boldsymbol{\alpha}^k$ and them is d is called the recall probability of $\boldsymbol{\alpha}^k$ from the Hamming distance d, and is denoted by $P(\boldsymbol{\alpha}^k, d)$. Moreover, the average of $P(\boldsymbol{\alpha}^k, d)$ over all the prototype vectors $\boldsymbol{\alpha}^1, \boldsymbol{\alpha}^2, \ldots, \boldsymbol{\alpha}^m$ is referred to as the average recall probability from Hamming distance d, and is denoted by $P_{\mathrm{av}}(d)$.

The prototype vectors used in this experiment are 26 English capital letters shown in Fig. 1. Each pattern has 49 $(= 7 \times 7)$ pixels where black and white pixels represent $+1$ and -1, respectively. For each value of the radius of neighborhood ($r = 1$, 2 and 3), we have determined the network parameters by applying Park's method and the proposed method to the prototype vectors. In both design methods, GEVM problems were solved by using the function "`gevp`" in MATLAB LMI Control Toolbox [20] with the same values of constants $L = 1$, $U = 10$ and $\varepsilon = 0.1$.

The simulation results are shown in Table 1. Here we have estimated the average recall probability $P_{\mathrm{av}}(d)$, $d = 1, 2, \ldots, 5$ by investigating the final output for all possible initial states in the case where $d \leq 2$ and by investigating the final output for randomly chosen 3,000 vectors from the possible initial states for each prototype vector in the case where $d \geq 3$. As shown in Table 1, the average recall probability of the proposed method is higher than that of Park's method in all cases. From these results, we can conclude that the proposed method is in fact superior to Park's method and therefore regarded as an efficient CNN design method for associative memories. Also, as shown in Table 1, the smaller the radius of neighborhood, the bigger the difference between the proposed method and Park's method. This is because the smaller the value of r, the more frequently the case where (15) is not feasible happens.

VI. Concluding Remarks

We have developed a CNN design method for associative memories by modifying the optimization-based method recently proposed by Park *et al.* with some analytical results presented in this paper. The proposed method tries to maximize the basins of attraction of the prototype vectors in any case, while the method proposed by Park *et al.* tries just to store the prototype vectors as memory vectors if a certain condition is not satisfied. As the computer simulation results show, the proposed method can achieve higher recall probability in all cases, and is effective in particular for CNNs with a small neighborhood.

We finally give some comments on the circuit implementation and the robustness of the proposed method. In the implementation of a CNN on an analog chip, the values of the parameters obtained numerically by the proposed method are not realized exactly due to inherent inaccuracy of analog devices. It is thus expected that the average recall probability achieved by a CNN chip is lower than that derived by computer simulations. However, since the proposed method chooses the values of the parameters so that κ_i in (14) or (17) is maximized, the small deviation of the parameter values will not affect those inequalities, that is, there will still exist a positive κ_i satisfying (14) or (17). In this sense, the proposed method is robust for parameter deviations. Moreover, it is obvious that the proposed method has a higher degree of robustness than Park's method. Quantitative investigation of the effect of the parameter deviation on the average recall probability is a future problem.

References

[1] L.O. Chua and L. Yang, "Cellular neural networks: theory," *IEEE Trans. Circuits Syst. I*, vol.35, pp.1257–1272, Oct. 1988.

[2] J.A. Nossek, "Design and learning with cellular neural networks," *Int. J. Circuit Theory Appl.*, vol.24, pp.15–24, Jan.–Feb. 1996.

[3] Á. Zarándy, "The art of CNN template design," *Int. J. Circuit Theory Appl.*, vol.27, pp.5–23, Jan.–Feb. 1999.

[4] M. Hänggi and G.S. Moschytz, "An exact and direct analytical method for the design of optimally robust CNN templates," *IEEE Trans. Circuits Syst. I*, vol.46, pp.304–311, Feb. 1999.

[5] M. Gilli and P.P. Civalleri, "Template design methods for binary stable cellular neural networks," *Int. J. Circuit Theory Appl.*, vol.30, pp.211–230, Mar.–Jun. 2002.

[6] J.J. Hopfield, "Neural networks and physical systems with emergent collective computational abilities," *Proc. Natl. Acad. Sci., U.S.A.*, vol.79, pp.2554–2558, Apr. 1982.

[7] L. Personnaz, I. Guyon, and G.Dreyfus, "Information storage and retrieval in spin-glass like neural networks," *J. Phys. Lett.*, vol.46, pp.L359–L365, Apr. 1985.

[8] L. Personnaz, I. Guyon, and G.Dreyfus, "Collective computational properties of neural networks: New learning mechanisms," *Phys. Lev. A*, vol.34, pp.4217–4228, Nov. 1986.

[9] Y. Kamp and M. Hasler, *Recursive Neural Networks for Associative Memory*, Chichester: John Wiley & Sons, 1990.

[10] A.N. Michel and J.A. Farrell, "Associative Memories via Artificial Neural Networks," *IEEE Control Syst. Mag.*, vol.10, pp.6–17, Apr. 1990.

[11] A.N. Michel and D. Liu, *Qualitative Analysis and Synthesis of Recurrent Neural Networks*, New York: Marcel Dekker, 2002.

[12] D. Liu and A.N. Michel, "Sparsely interconnected neural networks for associative memories with applications to cellular neural networks," *IEEE Trans. Circuits Syst. II*, vol.41, pp.295–307, Apr. 1994.

[13] G. Seiler, A.J. Schuler, and J.A. Nossek, "Design of robust cellular neural networks," *IEEE Trans. Circuits Syst. I*, vol.40, pp.358–364, May 1993.

[14] G. Grassi, "A new approach to design cellular neural networks for associative memories," *IEEE Trans. Circuits Syst. I*, vol.44, pp.835–838, Sep. 1997.

[15] G. Grassi, "On discrete-time cellular neural networks for associative memories," *IEEE Trans. Circuits Syst. I*, vol.48, pp.107–111, Jan. 2001.

[16] J. Park, H.-Y. Kim, Y. Park, and S.-W. Lee, "A synthesis procedure for associative memories based on space-varying cellular neural networks," *Neural Networks*, vol.14, no.1, pp.107–113, Jan. 2001.

[17] S. Boyd and L. El Ghaoui, "Method of centers for minimizing generalized eigenvalues," *Linear Algebra and its Applications*, vol.188, pp.63–111, Jul. 1993.

[18] Yu.E. Nesterov and A.S. Nemirovskii, "An interior-point method for generalized linear-fractional programming," *Mathematical Programming*, vol.69, pp.177–204, Jul. 1995.

[19] L. Vandenberghe and V. Balakrishnan, "Algorithms and software for LMI problems in control," *IEEE Control Syst. Mag.*, vol.17, pp.89–95, Oct. 1997.

[20] P. Gahinet, A. Nemirovski, A.J. Laub, and M. Chilali, *LMI control toolbox*, Natick: The MathWorks Inc., 1995.

[21] J.A.K. Suykens, J.P.L. Vandewalle, and B.L.R. De Moor, *Artificial neural networks for modelling and control of non-linear systems*, Kluwer Academic Publishers, Dordrecht, 1996.

[22] J.A.K. Suykens, B. De Moor, and J. Vandewalle, "Robust local stability of multilayer recurrent neural networks," *IEEE Trans. Neural Networks*, vol.11, pp.222–229, 2000.

[23] L.L.H. Andrew, "On binary output of cellular neural networks," *Int. J. Circuit Theory Appl.*, vol.25, pp.147–149, Mar.–Apr. 1997.

[24] L.O. Chua and C.W. Wu, " On the universe of stable cellular neural networks," *Int. J. Circuit Theory Appl.*, vol.20, pp.497–517, Spt.–Oct. 1992.

[25] M. Gilli, "Stability of cellular neural networks and delayed cellular neural networks with nonpositive templates and nonmonotonic output functions," *IEEE Trans. Circuits Syst. I*, vol.41, pp.518–528, Aug. 1994.

[26] N. Takahashi and L.O. Chua, "On the complete stability of nonsymmetric cellular neural networks," *IEEE Trans. Circuits Syst. I*, vol.45, pp.754–758, Jul. 1998.