Neural Networks 70 (2015) 9–17

Contents lists available at ScienceDirect

Neural Networks

journal homepage: www.elsevier.com/locate/neunet

Realization problem of multi-layer cellular neural networks

Jung-Chao Ban [a], Chih-Hung Chang [b,*]

[a] Department of Applied Mathematics, National Dong Hwa University, Hualien 970003, Taiwan, ROC

[b] Department of Applied Mathematics, National University of Kaohsiung, Kaohsiung 81148, Taiwan, ROC

ARTICLE INFO

Article history:
Received 25 November 2014
Received in revised form 13 April 2015
Accepted 12 June 2015
Available online 22 June 2015

Keywords:
Multi-layer cellular neural networks
Sofic shifts
Learning problem
Covering space
Separation property
Topological entropy

ABSTRACT

This paper investigates whether the output space of a multi-layer cellular neural network can be realized via a single layer cellular neural network in the sense of the existence of finite-to-one map from one output space to the other. Whenever such realization exists, the phenomena exhibited in the output space of the revealed single layer cellular neural network is at most a constant multiple of the phenomena exhibited in the output space of the original multi-layer cellular neural network. Meanwhile, the computation complexity of a single layer system is much less than the complexity of a multi-layer system. Namely, one can trade the precision of the results for the execution time. We remark that a routine extension of the proposed methodology in this paper can be applied to the substitution of hidden spaces although the detailed illustration is omitted.

1. Introduction

Cellular neural networks (CNNs), introduced by Chua and Yang (1988a, 1988b), have been one of the most investigated paradigms in neural information processing (Chua, 1998). CNNs must be completely stable in a wide range of applications (e.g., pattern recognition), i.e., each trajectory should converge toward some stationary state. The study of stationary solutions is thus important. Moreover, the investigation of *mosaic solutions* is essential due to the importance of learning algorithms and the training process. Roughly speaking, a learning algorithm is more efficient if there are more abundant output patterns for a given CNN.

Coupled systems based on CNNs, namely *multi-layer cellular neural networks* (MCNNs), have received considerable attention and have been successfully applied to many areas such as signal propagation between neurons, image processing, pattern recognition, information technology, CMOS realization and VLSI implementation (Arena, Baglio, Fortuna, & Manganaro, 1998; Ban & Chang, 2013; Carmona, Jimenez-Garrido, Dominguez-Castro, Espejo, & Rodriguez-Vazquez, 2002; Chua & Roska, 2002; Chua & Shi, 1991; Chua & Yang, 1988a; Crounse & Chua, 1995; Crounse, Roska, & Chua, 1993; Li, 2009; Murugesh, 2010; Peng, Zhang, & Liao, 2009; Xavier-de Souza, Yalcin, Suykens, & Vandewalle, 2004; Yang, Nishio, & Ushida, 2001, 2002). The development of CNNs has been inspired by the visual systems of mammals (Fukushima, 2013a, 2013b). The sufficient conditions for the complete stability of MCNNs can be found in Török and Roska (2004). Just as with CNNs, the study of mosaic solutions is also important and interesting. Recently, Ban and Chang (2009) showed that for MCNNs, more layers infer that models are capable of more phenomena.

A multi-layer cellular neural network is represented as

$$\begin{cases} \dfrac{d}{dt}x_i^{(N)}(t) = -x_i^{(N)}(t) + z^{(N)} \\ \qquad + \displaystyle\sum_{k\in\mathcal{N}}(a_k^{(N)}f(x_{i+k}^{(N)}(t)) + b_k^{(N)}f(x_{i+k}^{(N-1)}(t))), \\ \vdots \\ \dfrac{d}{dt}x_i^{(2)}(t) = -x_i^{(2)}(t) + z^{(2)} \\ \qquad + \displaystyle\sum_{k\in\mathcal{N}}(a_k^{(2)}f(x_{i+k}^{(2)}(t)) + b_k^{(2)}f(x_{i+k}^{(1)}(t))), \\ \dfrac{d}{dt}x_i^{(1)}(t) = -x_i^{(1)}(t) + z^{(1)} + \displaystyle\sum_{k\in\mathcal{N}}a_k^{(1)}f(x_{i+k}^{(1)}(t)), \end{cases} \tag{1}$$

for some integer $N \geq 2$, $i \in \mathbb{Z}$, and $t \geq 0$. The inputs for the neurons in the kth layer are the outputs of the $(k-1)$th layer in the proposed model (1) for $2 \leq k \leq N$. Fig. 1 shows the connections of a three-layer CNN with the nearest neighborhood. The so-called *neighborhood* $\mathcal{N}$ is a finite subset of integers $\mathbb{Z}$; the output function

$$f(x) = \frac{1}{2}(|x+1| - |x-1|) \tag{2}$$

* Corresponding author.
E-mail addresses: jcban@mail.ndhu.edu.tw (J.-C. Ban), chchang@nuk.edu.tw (C.-H. Chang).

http://dx.doi.org/10.1016/j.neunet.2015.06.003
0893-6080/© 2015 Elsevier Ltd. All rights reserved.

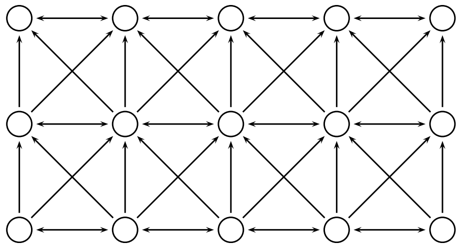

Fig. 1. Three-layer cellular neural networks with nearest neighborhood.

is a piecewise linear map. $\mathbb{A} = [A_1, \ldots, A_N]$ and $\mathbb{B} = [B_2, \ldots, B_N]$ are the feedback and controlling templates, respectively, where $A_j = [a_k^{(j)}]_{k\in\mathcal{N}}, B_l = [b_k^{(l)}]_{k\in\mathcal{N}}$ for $1 \le j \le N, 2 \le l \le N$; $\mathbb{z} = [z^{(1)}, \ldots, z^{(N)}]$ is the threshold. The template $\mathbb{T}$ of (1) consists of the feedback and controlling templates and the threshold, namely $\mathbb{T} = [\mathbb{A}, \mathbb{B}, \mathbb{z}]$. Note that (1) is a standard CNN if $N = 1$. This type of case is referred to as a *single layer CNN*.

A *mosaic solution* $\bar{x}$ is a solution of (1) which satisfies $|\bar{x}_i| > 1$ and its corresponding pattern $\bar{y} = (\bar{y}_i) = (f(\bar{x}_i))$ is called a *mosaic output pattern*. Since the output function (2) is piecewise linear with $f(x) = 1$ (resp. -1) if $x \ge 1$ (resp. $x \le -1$), the output of a mosaic solution $\bar{x} = (\bar{x}_i)_{i\in\mathbb{Z}}$ must be an element in $\Sigma = \{-1, +1\}^{\mathbb{Z}}$, which is why we call them *patterns*.

Given an N-layer MCNN with $N \ge 2$, we denote $\mathbf{Y}^{(N)}$ as the output solution space corresponding to a given input $(u_i)_{i\in\mathbb{Z}}$; namely,

$$\mathbf{Y}^{(N)} = \{(y_i)_{i\in\mathbb{Z}} : (y_i)_{i\in\mathbb{Z}} \text{ is a output solution for some input } (u_i)_{i\in\mathbb{Z}}\}.$$

A natural question arises: *when $Y^{(N)}$ behaves like a single layer CNN, can we find some map which links both systems*? Such a problem is called a *realization problem* since once this phenomenon has occurred, the given MCNN is realized by a single layer CNN system. From a mathematical point of view, the advantage of realizing a MCNN by a CNN is that one can *classify* MCNNs in terms of CNNs. If it indeed behaves like a single layer CNN, then it is easier to control. From an engineering perspective meanwhile, a realizable MCNN helps us to reduce the computation of such machines. In other words, this type of CNN is indeed a *shallow architecture* neural network model (cf. Bengio, 2009, Bengio & LeCun, 2007, Chang, 2015, Fukushima, 2013b, Hinton, Osindero, & Teh, 2006, Utgoff & Stracuzzi, 2002). Let $C(\Sigma, \Sigma)$ denote the set of continuous maps from Σ to Σ. A map $\tau \in C(\Sigma, \Sigma)$ is called *a factor* (resp. *an embedding*) if it is onto (resp. one-to-one). τ is called a conjugacy if it is both a factor and embedding. A realization problem can be achieved by raising the following problem.

Problem 1. Let $\mathbf{Y}^{(N)}$ be the output solution space of a MCNN (1) with $N \ge 2$.

(1) Does a pair $(\mathbf{Y}, \pi) \in \Sigma \times C(\Sigma, \Sigma)$ exist, where $\mathbf{Y}$ is the mosaic solution space of a single layer CNN and $\pi : \mathbf{Y} \to \mathbf{Y}^{(N)}$ is a factor from $\mathbf{Y}$ to $\mathbf{Y}^{(N)}$?
(2) Does π preserve the topological entropy, i.e., $h_{top}(\mathbf{Y}) = h_{top}(\pi(\mathbf{Y}))$?
(3) When does the factor π become an embedding, i.e., π is one-to-one?

Note that $\mathbf{Y}$ and $\mathbf{Y}^{(N)}$ are conjugate once (1) and (3) are satisfied.

It is worth pointing out that if π exists in the Problem 1–(1), then it links $\mathbf{Y}^{(N)}$ with some single layer CNN. Therefore, the output space $\mathbf{Y}^{(N)}$ is *controlled* by the factors π and $\mathbf{Y}$. If Problem 1-(2) holds, then the complexity of both $\mathbf{Y}^{(N)}$ and $\mathbf{Y}$ are the same, and it is important for the application of the learning algorithm. Finally, $\mathbf{Y}^{(N)}$ and $\mathbf{Y}$ are topologically the same if one ensures the factor π is also an embedding (Problem 1-(3)), and one simply replaces $\mathbf{Y}^{(N)}$ with $\mathbf{Y}$ in this case.

The aim of this paper is to study the above problem. Theorem 4.2 provides a natural and intrinsic characterization of Problem 1-(1) and 1-(2) by using the hidden Markov technique of symbolic dynamics. However, Problem 1-(3) is still unknown and is beyond the scope of the current study.

We also emphasize that one may raise the same problems on $\mathbf{Y}^{(i)}$ and $\mathbf{Y}^{(j)}$ for $1 \le i, j \le N$. More precisely, does a factor π between $\mathbf{Y}^{(i)}$ and $\mathbf{Y}^{(j)}$ exist for some $1 \le i, j \le n$ which preserves topological entropy? When does the factor π become an embedding? Some partial results are provided by Chang (2015). We emphasize that such a problem has to do with one given MCNN, and discuss the relationship between the output spaces of certain layers in such MCNNs. On the contrary, this study focuses on the relationship between a MCNN with other single layer CNNs. These two problems are different due to the fact that if $\mathbf{Y}^{(i)}$ and $\mathbf{Y}^{(j)}$ are extracted from the same MCNN, they inherit the same system information, making the discussion easier.

The rest of this paper is organized as follows. Section 2 considers the learning problem of two-layer cellular neural networks in pattern formation. Section 3 focuses on the realization problem of two-layer cellular neural networks, and the necessary and sufficient conditions for the existence of an entropy-preserving map between the output spaces of one and two-layer cellular neural networks. Following the discussion in Sections 2 and 3, Section 4 extends the results to general multi-layer cellular neural networks. Some discussion and suggestions for possible future research are given in Section 5 as a conclusion to the present work.

2. Learning problem of two-layer cellular neural networks

Learning problems (also called inverse problems) are some of the most investigated topics in a variety of disciplines. From a mathematical point of view, determining whether a given collection of output patterns can be seen through a CNN/MCNN is essential for the study of learning problems. This section reveals the necessary and sufficient conditions for the capability of exhibiting the output patterns of single layer cellular neural networks. The discussion can be applied to the elucidation of general cases, has addressed in Section 4.

A two-layer cellular neural network is seen as

$$\begin{cases} \dfrac{dx_i^{(2)}}{dt} = -x_i^{(2)} + \displaystyle\sum_{|k|\le d} a_k^{(2)} y_{i+k}^{(2)} + \sum_{|\ell|\le d} b_\ell^{(2)} u_{i+\ell}^{(2)} + z^{(2)}, \\ \dfrac{dx_i^{(1)}}{dt} = -x_i^{(1)} + \displaystyle\sum_{|k|\le d} a_k^{(1)} y_{i+k}^{(1)} + \sum_{|\ell|\le d} b_\ell^{(1)} u_{i+\ell}^{(1)} + z^{(1)}, \end{cases} \quad (3)$$

for some $d \in \mathbb{N}$, and $u_i^{(2)} = y_i^{(1)}$ for $i \in \mathbb{Z}$; $\mathbb{N}$ represents the set of positive integers and $\mathbb{Z}$ denotes the set of integers. The prototype of (3) is

$$\frac{dx_i}{dt} = -x_i + \sum_{|k|\le d} a_k y_{i+k} + \sum_{|\ell|\le d} b_\ell u_{i+\ell} + z. \quad (4)$$

Here $A = [-a_d, \ldots, a_d]$, $B = [-b_d, \ldots, b_d]$ are called *feedback* and *controlling templates*, respectively; z is known as the *threshold*, and $y_i = f(x_i) = \frac{1}{2}(|x_i + 1| - |x_i - 1|)$ is the output of x_i. The quantity x_i represents the state of the cell at i for $i \in \mathbb{Z}$. The output of a stationary solution $\bar{x} = (\bar{x}_i)_{i\in\mathbb{Z}}$ is called an output pattern. A *mosaic solution* $\bar{x}$ satisfies $|\bar{x}_i| > 1$ and its corresponding pattern $\bar{y}$ is called a *mosaic output pattern*. Considering the mosaic solution $\bar{x}$,

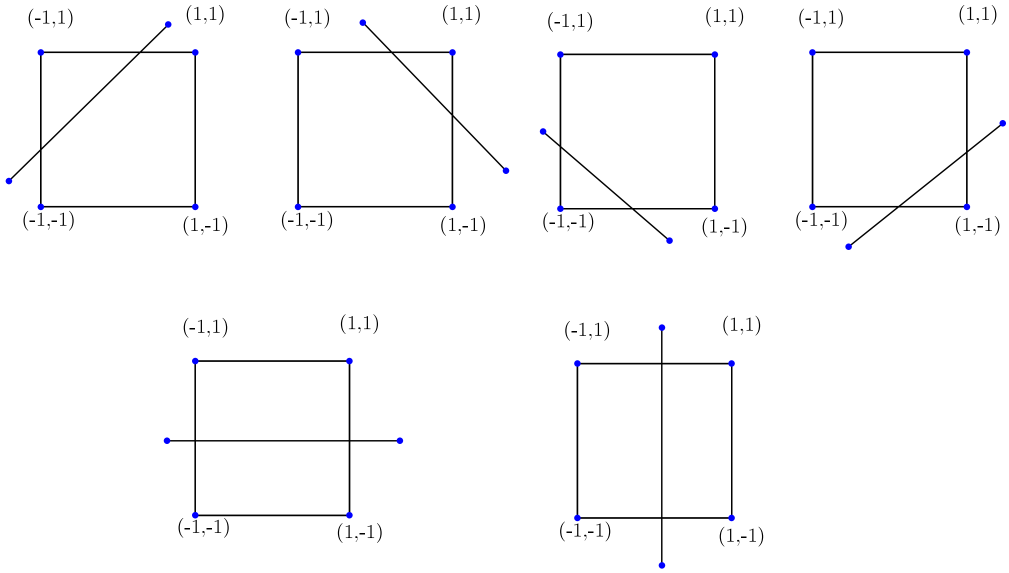

Fig. 2. Suppose U is a proper subset of $V^2 = \{-1, 1\}^2$. There are only 12 possible choice of U that satisfies linear separation property.

the necessary and sufficient conditions for the state "+" at cell C_i, i.e., $\bar{y}_i = 1$, is

$$a - 1 + z > -\left(\sum_{0<|k|\le d} a_k \bar{y}_{i+k} + \sum_{|\ell|\le d} b_\ell u_{i+\ell}\right), \tag{5}$$

where $a = a_0$. Similarly, the necessary and sufficient conditions for the state "−" at cell C_i, i.e., $\bar{y}_i = -1$, is

$$a - 1 - z > \sum_{0<|k|\le d} a_k \bar{y}_{i+k} + \sum_{|\ell|\le d} b_\ell u_{i+\ell}. \tag{6}$$

To avoid ambiguity we denote the notation $\bar{y}_i$ as y_i and refer to the output pattern $y_{-d} \cdots y_0 \cdots y_d$ coupled with the input $u_{-d} \cdots u_0 \cdots u_d$ as

$$\boxed{\begin{array}{l} y_{-d} \cdots y_{-1} y_0 y_1 \cdots y_d \\ u_{-d} \cdots u_{-1} u_0 u_1 \cdots u_d \end{array}}$$

$$\equiv y_{-d} \cdots y_d \diamond u_{-d} \cdots u_d \in \{-1, 1\}^{(2d+1)\times 2}. \tag{7}$$

Let

$$V^n = \{v \in \mathbb{R}^n : v = (v_1, v_2, \ldots, v_n), \text{ and } |v_i| = 1, 1 \le i \le n\},$$

where $n = 4d + 1$. (5) and (6) can be rewritten in a compact form by introducing the following notation.

Set $\alpha = (a_{-d}, \ldots, a_{-1}, a_1, \ldots, a_d)$ and $\beta = (b_{-d}, \ldots, b_d)$. Then, α can be used to represent A', the surrounding template of A without a center, and β can be used to represent the template B. The basic set of admissible local patterns with the "+" state in the center is defined as

$$\mathcal{B}_+(A, B, z) = \{v \diamond w : (v, w) \in V^n,$$
$$a - 1 + z > -(\alpha \cdot v + \beta \cdot w)\}, \tag{8}$$

where "·" is the inner product in Euclidean space. Similarly, the basic set of admissible local patterns with the "−" state in the center is defined as

$$\mathcal{B}_-(A, B, z) = \{v \diamond w : (v, w) \in V^n,$$
$$a - 1 - z > \alpha \cdot v + \beta \cdot w\}. \tag{9}$$

Furthermore, the admissible local patterns induced by (A, B, z) can be denoted by

$$\mathcal{B}(A, B, z) = (\widetilde{\mathcal{B}}_+(A, B, z), \widetilde{\mathcal{B}}_-(A, B, z)), \tag{10}$$

where

$$\widetilde{\mathcal{B}}_+(A, B, z) = \{v_+ \diamond w : v \diamond w \in \mathcal{B}_+(A, B, z)\},$$
$$\widetilde{\mathcal{B}}_-(A, B, z) = \{v_- \diamond w : v \diamond w \in \mathcal{B}_-(A, B, z)\},$$

and $v_+ \in V^{2d+1}$ (resp. $v_- \in V^{2d+1}$) are obtained by inserting 1 (resp. -1) at the center coordinate of $v \in V^{2d}$.

Suppose U is a subset of V^n. Let $U^c = V^n \setminus U$. We say that U satisfies the *linear separation property* if there exists a hyperplane H that separates U and U^c. More precisely, U satisfies the separation property if and only if there exists a linear functional $g(z_1, z_2, \ldots, z_n) = c_1 z_1 + c_2 z_2 + \cdots + c_n z_n$ such that

$$g(v) > 0 \quad \text{for } v \in U \quad \text{and} \quad g(v) < 0 \quad \text{for } v \in U^c.$$

Fig. 2 interprets those $U \subset V^2$ satisfying the linear separation property.

Let

$$\mathcal{P}^{n+2} = \{(A, B, z) | A, B \in \mathcal{M}_{1\times(2d+1)}(\mathbb{R}), z \in \mathbb{R}\}, \tag{11}$$

where $n = 4d+1$ and $\mathcal{M}_{r\times s}(\mathbb{R})$ indicates a $r \times s$ real-valued matrix. Ban, Chang, Lin, and Lin (2009) showed that the parameter space $\mathcal{P}^{n+2}$ can be partitioned into finite equivalent sub-regions; that is, two sets of parameters induce identical basic sets of admissible local patterns if and only if they belong to the same partition in the parameter space.

Theorem 2.1 (*See* Hsu, Juang, Lin, & Lin, 2000). *There exists a positive integer $K(n)$ and a unique collection of open subsets $\{P_k\}_{k=1}^K$ of $\mathcal{P}^{n+2}$ satisfying*

(i) $\mathcal{P}^{n+2} = \cup_{k=1}^K \overline{P}_k$.
(ii) $P_k \cap P_\ell = \varnothing$ *for all* $k \ne \ell$.
(iii) $\mathcal{B}(A, B, z) = \mathcal{B}(A', B', z')$ *if and only if* $(A, B, z), (A', B', z') \in P_k$ *for some k.*

Here $\overline{P}$ indicates the closure of P in $\mathcal{P}^{n+2}$.

Remark 2.2. Ban et al. extended Theorem 2.1 to the case where the parameter space of a MCNN can also be partitioned into finite equivalent sub-regions (Ban et al., 2009). More specifically, Theorem 2.1 plays an essential role in the learning problem of MCNNs. For each sub-region $P \subset \mathcal{P}^{n+2}$, every collection of corresponding parameters is obtained from the coefficients of the equation of a hyperplane H that separates not only $\mathcal{B}_+(A, B, z)$ and $\mathcal{B}_+(A, B, z)^c$ but also $\mathcal{B}_-(A, B, z)$ and $\mathcal{B}_-(A, B, z)^c$. Readers are referred to Ban and Chang (2009) and Ban et al. (2009) for more details.

Theorem 2.1, as mentioned in Remark 2.2, "implicitly" demonstrates that a collection of patterns $\mathcal{B} = (\widetilde{\mathcal{B}}_+, \widetilde{\mathcal{B}}_-)$ is realized via a cellular neural network, i.e., there exists A, B and z such that $\widetilde{\mathcal{B}}_+(A, B, z) = \widetilde{\mathcal{B}}_+$ and $\widetilde{\mathcal{B}}_-(A, B, z) = \widetilde{\mathcal{B}}_-$, if and only if $\mathcal{B}_+, \mathcal{B}_+^c$ and $\mathcal{B}_-, \mathcal{B}_-^c$ can be separated by same hyperplane, respectively. Theorem 2.3 enhances the result of Theorem 2.1 and provides an affirmative response for the inverse problem of MCNNs explicitly. Before presenting the theorem we consider the following example. Suppose $U \subseteq V^n$. Let $-U$ denote the reflection of U with respect to the origin, or in other words,

$$-U = \{(v_1, v_2, \ldots, v_n) \in V^n : (-v_1, -v_2, \ldots, -v_n) \in U\}.$$

Theorem 2.3. *A collection of patterns $\mathcal{B} = (\widetilde{\mathcal{B}}_+, \widetilde{\mathcal{B}}_-)$ can be realized in (4) if and only if either of the following conditions is satisfied:*

(Inv1) $-\mathcal{B}_+ \subseteq \mathcal{B}_-$ *and $\mathcal{B}_-$ satisfies linear separation property.*
(Inv2) $-\mathcal{B}_- \subseteq \mathcal{B}_+$ *and $\mathcal{B}_+$ satisfies linear separation property;*

Proof. It suffices to show that both (Inv1) and (Inv2) infer that there exists A, B, and z such that $\mathcal{B} = \mathcal{B}(A, B, z)$. As follows, we show that $\mathcal{B}$ satisfies (Inv1) and is adequate for realizing $\mathcal{B}$ in (4). Other cases can be performed analogously.

Since $\mathcal{B}_-$ satisfies the linear separation property, there exist parameters

$$a_1, a_2, \ldots, a_{2d}, b_1, b_2, \ldots, b_{2d+1}, a, \text{ and } z$$

such that

(i) the hyperplane

$$H : a - z - 1 - \sum_{i=1}^{2d} a_i t_i + \sum_{j=1}^{2d+1} b_j s_j = 0$$

separates $\mathcal{B}_-$ and $\mathcal{B}_-^c$;

(ii) $g(v_1, w_1) \neq g(v_2, w_2)$ for all $(v_1, w_1) \neq (v_2, w_2) \in V^n$, where $g(v, w) = \alpha \cdot v + \beta \cdot w$, $\alpha = (a_1, a_2, \ldots, a_{2d})$, and $\beta = (b_1, b_2, \ldots, b_{2d+1})$.

Without loss of generality, we may assume that

$$\begin{cases} a - z > 1 + g(v, w), & \text{if } v \diamond w \in \mathcal{B}_-; \\ a - z < 1 + g(v, w), & \text{if } v \diamond w \in \mathcal{B}_-^c. \end{cases}$$

Let

$$c_1 = \max\{g(v, w) : v \diamond w \in \mathcal{B}_-\},$$
$$c_2 = \min\{g(v, w) : v \diamond w \in \mathcal{B}_-^c\},$$
$$c_3 = \max\{g(v, w) : v \diamond w \in -\mathcal{B}_+\},$$
$$c_4 = \min\{g(v, w) : v \diamond w \in (-\mathcal{B}_+)^c\}.$$

Then

$$c_1 = c_3 < c_2 = c_4 \quad \text{if } -\mathcal{B}_+ = \mathcal{B}_-;$$
$$c_3 < c_4 \leq c_1 < c_2 \quad \text{if } -\mathcal{B}_+ \subsetneq \mathcal{B}_-.$$

Let

$$\bar{a} = 1 + \frac{1}{4}(c_1 + c_2 + c_3 + c_4) \quad \text{and} \quad \bar{z} = \frac{1}{4}(c_3 + c_4 - c_1 - c_2).$$

It follows that

$$\bar{a} - 1 + \bar{z} > -g(v, w) \quad \text{for all } v \diamond w \in \mathcal{B}_+;$$
$$\bar{a} - 1 + \bar{z} < -g(v, w) \quad \text{for all } v \diamond w \in \mathcal{B}_+^c$$

and

$$\bar{a} - 1 - \bar{z} > g(v, w) \quad \text{for all } v \diamond w \in \mathcal{B}_-;$$
$$\bar{a} - 1 - \bar{z} < g(v, w) \quad \text{for all } v \diamond w \in \mathcal{B}_-^c.$$

In other words, let $\overline{A}$ be obtained from A by replacing a as $\bar{a}$. Then $\mathcal{B}(\overline{A}, B, \bar{z}) = \mathcal{B}$.

This completes the proof. □

3. Realizing two-layer cellular neural networks via cellular neural networks

This section elucidates the realization problem of two-layer cellular neural networks in order to clarify methodology behind general MCNNs, as presented in the next section. More specifically, we investigate if the output space of a given two-layer cellular neural network can be embedded into the output space of some single layer cellular neural network with a finite-to-one map. Notably, the methodology we introduce also works for the realization problem with respect to hidden space.

To make the discussion easier to understand, we consider a *simplified two-layer cellular neural networks* (STCNNs) proposed as

$$\begin{cases} \dfrac{dx_i^{(2)}}{dt} = -x_i^{(2)} + a^{(2)} y_i^{(2)} + a_r^{(2)} y_{i+1}^{(2)} \\ \qquad + b^{(2)} u_i^{(2)} + b_r^{(2)} u_{i+1}^{(2)} + z^{(2)}, \\ \dfrac{dx_i^{(1)}}{dt} = -x_i^{(1)} + a^{(1)} y_i^{(1)} + a_r^{(1)} y_{i+1}^{(1)} + z^{(1)}, \end{cases} \tag{12}$$

where $u_i^{(2)} = y_i^{(1)}$ for all $i \in \mathbb{Z}$. Suppose $\mathbf{y} = \begin{pmatrix} \cdots y_{-1}^{(2)} y_0^{(2)} y_1^{(2)} \cdots \\ \cdots y_{-1}^{(1)} y_0^{(1)} y_1^{(1)} \cdots \end{pmatrix}$ is a mosaic pattern. For $i \in \mathbb{Z}$, $y_i^{(1)} = 1$ if and only if

$$a^{(1)} + z^{(1)} - 1 > -a_r^{(1)} y_{i+1}^{(1)}. \tag{13}$$

Similarly, $y_i^{(1)} = -1$ if and only if

$$a^{(1)} - z^{(1)} - 1 > a_r^{(1)} y_{i+1}^{(1)}. \tag{14}$$

The same argument asserts

$$a^{(2)} + z^{(2)} - 1 > -a_r^{(2)} y_{i+1}^{(2)} - (b^{(2)} u_i^{(2)} + b_r^{(2)} u_{i+1}^{(2)}), \tag{15}$$

and

$$a^{(2)} - z^{(2)} - 1 > a_r^{(2)} y_{i+1}^{(2)} + (b^{(2)} u_i^{(2)} + b_r^{(2)} u_{i+1}^{(2)}) \tag{16}$$

are the necessary and sufficient conditions for $y_i^{(2)} = -1$ and $y_i^{(2)} = 1$, respectively. Note that the quantity $u_i^{(2)}$ in (15) and (16) satisfies $|u_i^{(2)}| = 1$ for each i. Define $\xi_1 : \{-1, 1\} \to \mathbb{R}$ and $\xi_2 : \{-1, 1\}^{3\times 1} \to \mathbb{R}$ by

$$\xi_1(w) = a_r^{(1)} w, \qquad \xi_2(w_1, w_2, w_3) = a_r^{(2)} w_1 + b^{(2)} w_2 + b_r^{(2)} w_3.$$

Set

$$\mathcal{B}^{(1)} = \left\{ \boxed{y^{(1)} y_r^{(1)}} : y^{(1)}, y_r^{(1)} \in \{-1, 1\} \text{ satisfy (13) and (14)} \right\},$$

$$\mathcal{B}^{(2)} = \left\{ \boxed{\begin{matrix} y^{(2)} y_r^{(2)} \\ u^{(2)} u_r^{(2)} \end{matrix}} : y^{(2)}, y_r^{(2)}, u^{(2)}, u_r^{(2)} \in \{-1, 1\} \right.$$
$$\left. \text{satisfy (15) and (16)} \right\}.$$

That is,

$$\boxed{y^{(1)} y_r^{(1)}} \in \mathcal{B}^{(1)} \Leftrightarrow \begin{cases} a^{(1)} + z^{(1)} - 1 > -\xi_1(y_r^{(1)}), & \text{if } y^{(1)} = 1; \\ a^{(1)} - z^{(1)} - 1 > \xi_1(y_r^{(1)}), & \text{if } y^{(1)} = -1. \end{cases}$$

$$\boxed{\begin{matrix} y^{(2)} y_r^{(2)} \\ u^{(2)} u_r^{(2)} \end{matrix}} \in \mathcal{B}^{(2)}$$
$$\Leftrightarrow \begin{cases} a^{(2)} + z^{(2)} - 1 > -\xi_2(y_r^{(2)}, u^{(2)}, u_r^{(2)}), & \text{if } y^{(2)} = 1; \\ a^{(2)} - z^{(2)} - 1 > \xi_2(y_r^{(2)}, u^{(2)}, u_r^{(2)}), & \text{if } y^{(2)} = -1. \end{cases}$$

The basic set of admissible local patterns $\mathcal{B}$ of (12) is then

$$\mathcal{B} = \left\{ \boxed{\begin{matrix} yy_r \\ uu_r \end{matrix}} : \boxed{\begin{matrix} yy_r \\ uu_r \end{matrix}} \in \mathcal{B}^{(2)} \text{ and } \boxed{uu_r} \in \mathcal{B}^{(1)} \right\}.$$

The basic set of admissible local patterns plays an essential role in investigating the structure of the solution space $\mathbf{Y}$ of STCNNs. Substitute mosaic patterns -1 and 1 as symbols $-$and $+$, respectively. Define the ordering matrix of $\{-,+\}^{2\times 2}$ by

$$\mathbb{X}_2 = \begin{array}{c} \\ \boxed{\begin{matrix} - \\ - \end{matrix}} \\ \boxed{\begin{matrix} - \\ + \end{matrix}} \\ \boxed{\begin{matrix} + \\ - \end{matrix}} \\ \boxed{\begin{matrix} + \\ + \end{matrix}} \end{array} \begin{array}{c} \begin{array}{cccc} \boxed{\begin{matrix} - \\ - \end{matrix}} & \boxed{\begin{matrix} - \\ + \end{matrix}} & \boxed{\begin{matrix} + \\ - \end{matrix}} & \boxed{\begin{matrix} + \\ + \end{matrix}} \end{array} \\ \begin{pmatrix} \boxed{\begin{matrix} -- \\ -- \end{matrix}} & \boxed{\begin{matrix} -- \\ -+ \end{matrix}} & \boxed{\begin{matrix} -+ \\ -- \end{matrix}} & \boxed{\begin{matrix} -+ \\ -+ \end{matrix}} \\ \boxed{\begin{matrix} -- \\ +- \end{matrix}} & \boxed{\begin{matrix} -- \\ ++ \end{matrix}} & \boxed{\begin{matrix} -+ \\ +- \end{matrix}} & \boxed{\begin{matrix} -+ \\ ++ \end{matrix}} \\ \boxed{\begin{matrix} +- \\ -- \end{matrix}} & \boxed{\begin{matrix} +- \\ -+ \end{matrix}} & \boxed{\begin{matrix} ++ \\ -- \end{matrix}} & \boxed{\begin{matrix} ++ \\ -+ \end{matrix}} \\ \boxed{\begin{matrix} +- \\ +- \end{matrix}} & \boxed{\begin{matrix} +- \\ ++ \end{matrix}} & \boxed{\begin{matrix} ++ \\ +- \end{matrix}} & \boxed{\begin{matrix} ++ \\ ++ \end{matrix}} \end{pmatrix} \end{array} := (x_{pq})_{1\le p,q\le 4}.$$

Notably each entry in $\mathbb{X}$ is a 2×2 pattern since $\mathcal{B}$ consists of 2×2 local patterns. Suppose that $\mathcal{B}$ is given. The *transition matrix* $\mathbf{T} \equiv \mathbf{T}(\mathcal{B}) \in \mathcal{M}_4(\{0,1\})$ is defined by

$$\mathbf{T}(p,q) = \begin{cases} 1, & \text{if } x_{pq} \in \mathcal{B}; \\ 0, & \text{otherwise.} \end{cases}$$

where $1 \le p, q \le 4$. Let $\mathcal{L} = \{\alpha_1, \alpha_2, \alpha_3, \alpha_4\}$, where

$$\alpha_1 := --, \qquad \alpha_2 = -+, \qquad \alpha_3 = +-, \qquad \alpha_4 = ++.$$

Define the *symbolic transition matrix* $\mathbf{S} \equiv \mathbf{S}(\mathcal{B})$ as

$$\mathbf{S}(p,q) = \begin{cases} \alpha_i, & \text{if } \mathbf{T}(p,q) = 1 \text{ and } x_{pq} = \alpha_i \diamond \alpha_j \text{ for some } j; \\ \varnothing, & \text{otherwise.} \end{cases} \tag{17}$$

In Ban and Chang (2013), Ban and Chang demonstrated that the structure of the *output space*

$$\mathbf{Y}^{(2)} = \{(y_i)_{i\in\mathbb{Z}} : \text{ there exists } (u_i)_{i\in\mathbb{Z}} \text{ such that } (y_i \diamond u_i)_{i\in\mathbb{Z}} \in \mathbf{Y}\}$$

is determined by $\mathbf{S}$. Similarly we can investigate the topological property of the *hidden space*

$$\mathbf{Y}^{(1)} = \{(u_i)_{i\in\mathbb{Z}} : \text{ there exists } (y_i)_{i\in\mathbb{Z}} \text{ such that } (y_i \diamond u_i)_{i\in\mathbb{Z}} \in \mathbf{Y}\}$$

with an appropriate symbolic transition matrix.

From a mathematical viewpoint, a realization problem investigates whether the output space of a two-layer system is topologically conjugate to the output space of a single layer system. To achieve the desired result, we recall the groundwork known as *Williams' Classification Theorem*.

Definition 3.1. Let A and B be nonnegative integral matrices. An *elementary equivalence* from A to B is a pair (R, S) of rectangular nonnegative matrices satisfying $A = RS$ and $B = SR$. A *strong shift equivalence* from A to B is a sequence of ℓ elementary equivalences

$$\begin{aligned} &(R_1, S_1) : A = A_0 = R_1S_1, \qquad A_1 = S_1R_1 \\ &(R_2, S_2) : A_1 = R_2S_2, \qquad A_2 = S_2R_2 \\ &\vdots \\ &(R_\ell, S_\ell) : A_{\ell-1} = R_\ell S_\ell, \qquad A_\ell = B = S_\ell R_\ell \end{aligned}$$

for some ℓ. In this case we say that A is strong shift equivalent to B and write $A \approx B$.

Theorem 3.2 (*Williams' Classification Theorem, See Williams, 1973*). *Suppose A and B are nonnegative integral matrices. Let X and Y be the spaces with transition matrices A and B, respectively. Then X and Y are topologically conjugate if and only if A and B possess strong shift equivalence.*

The realization problem of two-layer cellular neural networks begins with partitioning $\mathbf{S}$ into an ordered collection of symbolic matrices $\mathbf{S}_1, \mathbf{S}_2, \mathbf{S}_3, \mathbf{S}_4$, which was introduced by Coven and Paul (1975), as follows. For $i = 1, 2, 3, 4$, define $\mathbf{S}_i$ as

$$\mathbf{S}_i(p,q) = \begin{cases} \alpha_i, & \mathbf{S}(p,q) = \alpha_i; \\ \varnothing, & \text{otherwise.} \end{cases} \tag{18}$$

In other words, $\mathbf{S}_i$ is derived by projecting $\mathbf{S}$ with respect to the symbol α_i, and $\mathbf{S} = \mathbf{S}_1 + \mathbf{S}_2 + \mathbf{S}_3 + \mathbf{S}_4$. Following the definition of $\mathbf{S}_i$ we set a 0–1 matrix $\mathbf{T}_i$ as

$$\mathbf{T}_i(p,q) = \begin{cases} 1, & \mathbf{S}_i(p,q) \neq \varnothing; \\ 0, & \text{otherwise.} \end{cases} \tag{19}$$

Let $\{0, e\}$ be the trivial Boolean algebra. Construct $\mathbf{T}_i^*$, a matrix over $\{0, e\}$, from $\mathbf{T}_i$ by

$$\mathbf{T}_i^*(p,q) = e \Longleftrightarrow \mathbf{T}_i(p,q) = 1 \tag{20}$$

for $1 \le i \le 4$.

Furthermore, let $\mathbb{G}$ be the semigroup generated by $\{\mathbf{T}_i^*\}_{i=1}^4$. For $i = 1, 2, 3, 4$, $\mathbf{A}_i$ is a 0–1 matrix indexed by $(\mathbb{G} \setminus \{\mathbf{0}\}) \times \{1, 2, 3, 4\}$ which is defined as

$$\mathbf{A}_i((R,k),(S,\ell)) = \begin{cases} 1, & k = i \text{ and } R = \mathbf{T}_i^* S; \\ 0, & \text{otherwise} \end{cases} \tag{21}$$

where $\mathbf{0} \in \mathcal{M}_4(\{0, e\})$ is the zero matrix. Theorem 3.3 therefore asserts an affirmative answer for the realization problem of two-layer cellular neural networks.

Theorem 3.3. *Suppose $\mathbf{T}$ and $\mathbf{S}$ are the transition matrix and symbolic transition matrix of the output space $\mathbf{Y}^{(2)}$ of (12), respectively. Let $\mathbf{S}_i$, $\mathbf{T}_i$, $\mathbf{T}_i^*$, and $\mathbf{A}_i$ be defined as in (18), (19), (20), and (21), respectively, and let $\mathbf{A}$ be the square matrix obtained from $\mathbf{A}_1 + \mathbf{A}_2 + \mathbf{A}_3 + \mathbf{A}_4$ by deleting both the kth row and the kth column if either of which is a zero vector. Then there is a single layer cellular neural network with output space $\mathbf{Y}'$ and a finite-to-one factor $\Phi : \mathbf{Y}' \to \mathbf{Y}^{(2)}$ if and only if $\mathbf{A} \approx \mathbf{T}'$, where $\mathbf{T}'$ is the transition matrix of $\mathbf{Y}'$.*

Remarkably, Theorem 3.3 works for realizing the hidden space $\mathbf{Y}^{(1)}$ via the output space of a single layer cellular neural network with a minor modification. More precisely, the realization problem of hidden space can be elucidated by substituting the symbolic transition matrix $\mathbf{S}$ with $\mathbf{S}'$, where

$$\mathbf{S}'(p,q) = \begin{cases} \alpha_j, & \text{if } \mathbf{T}(p,q) = 1 \text{ and } x_{pq} = \alpha_i \diamond \alpha_j \text{ for some } i; \\ \varnothing, & \text{otherwise.} \end{cases} \tag{22}$$

Instead of addressing the proof of Theorem 3.3 immediately, the following example explores the key points of the existence of a single layer cellular neural network and its corresponding factor. Rigorous proof is postponed to the next section and is illustrated by a general case.

Consider a STCNN given by

$$\begin{cases} \dfrac{dx_i^{(2)}}{dt} = -x_i^{(2)} - 0.3y_i^{(2)} - 1.2y_{i+1}^{(2)} \\ \qquad + 0.7y_i^{(1)} + 2.3y_{i+1}^{(1)} + 0.9, \\ \dfrac{dx_i^{(1)}}{dt} = -x_i^{(1)} + 2.9y_i^{(1)} + 1.7y_{i+1}^{(1)} + 0.1. \end{cases} \tag{23}$$

It follows that the basic set of admissible local patterns of the mosaic solution space of (23) is

$$\mathcal{B} = \left\{ \boxed{\begin{matrix} -+ \\ -- \end{matrix}}, \boxed{\begin{matrix} -+ \\ +- \end{matrix}}, \boxed{\begin{matrix} +- \\ -+ \end{matrix}}, \boxed{\begin{matrix} +- \\ ++ \end{matrix}}, \boxed{\begin{matrix} ++ \\ -+ \end{matrix}}, \boxed{\begin{matrix} ++ \\ ++ \end{matrix}} \right\}.$$

The transition matrix $\mathbf{T}$ and the symbolic transition matrix $\mathbf{S}$ of the hidden space $\mathbf{Y}^{(1)}$ are

$$\mathbf{T} = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 \end{pmatrix} \quad \text{and} \quad \mathbf{S} = \begin{pmatrix} \varnothing & \varnothing & \alpha_1 & \varnothing \\ \varnothing & \varnothing & \alpha_3 & \varnothing \\ \varnothing & \alpha_2 & \varnothing & \alpha_2 \\ \varnothing & \alpha_4 & \varnothing & \alpha_4 \end{pmatrix},$$

respectively. Partitioning $\mathbf{S}$ and $\mathbf{T}$ as defined in (18) and (19) infers that

$$\mathbf{S}_1 = \begin{pmatrix} \varnothing & \varnothing & \alpha_1 & \varnothing \\ \varnothing & \varnothing & \varnothing & \varnothing \\ \varnothing & \varnothing & \varnothing & \varnothing \\ \varnothing & \varnothing & \varnothing & \varnothing \end{pmatrix}, \qquad \mathbf{S}_2 = \begin{pmatrix} \varnothing & \varnothing & \varnothing & \varnothing \\ \varnothing & \varnothing & \varnothing & \varnothing \\ \varnothing & \alpha_2 & \varnothing & \alpha_2 \\ \varnothing & \varnothing & \varnothing & \varnothing \end{pmatrix},$$

$$\mathbf{S}_3 = \begin{pmatrix} \varnothing & \varnothing & \varnothing & \varnothing \\ \varnothing & \varnothing & \alpha_3 & \varnothing \\ \varnothing & \varnothing & \varnothing & \varnothing \\ \varnothing & \varnothing & \varnothing & \varnothing \end{pmatrix}, \qquad \mathbf{S}_4 = \begin{pmatrix} \varnothing & \varnothing & \varnothing & \varnothing \\ \varnothing & \varnothing & \varnothing & \varnothing \\ \varnothing & \varnothing & \varnothing & \varnothing \\ \varnothing & \alpha_4 & \varnothing & \alpha_4 \end{pmatrix},$$

and

$$\mathbf{T}_1 = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}, \qquad \mathbf{T}_2 = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{pmatrix},$$

$$\mathbf{T}_3 = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}, \qquad \mathbf{T}_4 = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 \end{pmatrix}.$$

A routine computation that deduces the semigroup $\mathbb{G}$ generated by $\{\mathbf{T}_i^*\}_{i=1}^4$ is

$$\mathbb{G} = \{\mathbf{0}, \mathbf{T}_1^*, \mathbf{T}_2^*, \mathbf{T}_3^*, \mathbf{T}_4^*, \mathbf{T}_5^*, \mathbf{T}_6^*, \mathbf{T}_7^*, \mathbf{T}_8^*\},$$

where $\mathbf{T}_1^*, \mathbf{T}_2^*, \mathbf{T}_3^*, \mathbf{T}_4^*$ is defined in (20), and

$$\mathbf{T}_5^* = \begin{pmatrix} 0 & e & 0 & e \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}, \qquad \mathbf{T}_6^* = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & e & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix},$$

$$\mathbf{T}_7^* = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & e & 0 & e \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}, \qquad \mathbf{T}_8^* = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & e & 0 \end{pmatrix}.$$

In addition, $\mathbf{T}_i^*\mathbf{T}_j^* = \mathbf{0}$ for $1 \le i \le 4$, $1 \le j \le 8$ except

(1) $\mathbf{T}_1^*\mathbf{T}_2^* = \mathbf{T}_5^*$, $\mathbf{T}_1^*\mathbf{T}_6^* = \mathbf{T}_1^*$;
(2) $\mathbf{T}_2^*\mathbf{T}_3^* = \mathbf{T}_6^*$, $\mathbf{T}_2^*\mathbf{T}_4^* = \mathbf{T}_2^*$, $\mathbf{T}_2^*\mathbf{T}_7^* = \mathbf{T}_2^*$, $\mathbf{T}_2^*\mathbf{T}_8^* = \mathbf{T}_6^*$;
(3) $\mathbf{T}_3^*\mathbf{T}_2^* = \mathbf{T}_7^*$, $\mathbf{T}_3^*\mathbf{T}_6^* = \mathbf{T}_3^*$;
(4) $\mathbf{T}_4^*\mathbf{T}_3^* = \mathbf{T}_8^*$, $\mathbf{T}_4^*\mathbf{T}_4^* = \mathbf{T}_4^*$, $\mathbf{T}_4^*\mathbf{T}_7^* = \mathbf{T}_4^*$, $\mathbf{T}_4^*\mathbf{T}_8^* = \mathbf{T}_8^*$.

It comes immediately that

$$\mathbf{A}_1 = E_2 \otimes \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix},$$

$$\mathbf{A}_2 = E_2 \otimes \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix},$$

$$\mathbf{A}_3 = E_2 \otimes \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix},$$

$$\mathbf{A}_4 = E_2 \otimes \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \end{pmatrix},$$

where $E_2 = \left(\begin{smallmatrix} 1 & 1 \\ 1 & 1 \end{smallmatrix}\right)$ and $\otimes$ is the Kronecker product. We point out that $\mathbf{A}_i$ is indexed by

$(\mathbf{T}_1^*, 1), (\mathbf{T}_2^*, 1), \ldots, (\mathbf{T}_8^*, 1), (\mathbf{T}_1^*, 2), (\mathbf{T}_2^*, 2), \ldots, (\mathbf{T}_7^*, 4), (\mathbf{T}_8^*, 4)$.

After deleting those zero rows and columns from $\Sigma_{i=1}^4 \mathbf{A}_i$ the desired matrix $\mathbf{A}$ is seen as

$$\mathbf{A} = \begin{pmatrix} 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 \end{pmatrix}.$$

By rearranging the second and third indices of $\mathbf{A}$, to avoid ambiguity, we still denote the new matrix as $\mathbf{A}$, and thus we have

$$\mathbf{A} = I_2 \otimes \begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 1 \end{pmatrix}, \quad \text{where } I_2 \text{ is the } 2 \times 2 \text{ identity matrix.}$$

To restate, $\mathbf{A}$ is a diagonal matrix composed of two identical irreducible sub-matrices.

It remains to find a single layer cellular neural network with transition matrix $\mathbf{T}$ being strong shift equivalent to $\left(\begin{smallmatrix} 0 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 1 \end{smallmatrix}\right)$.

Consider $\mathbf{T} = \left(\begin{smallmatrix} 1 & 1 \\ 1 & 0 \end{smallmatrix}\right)$, which is the transition matrix of the cellular neural network

$$\frac{dx_i}{dt} = -x_i + 2y_i - y_{i+1} - 0.3, \quad i \in \mathbb{Z}. \tag{24}$$

Let

$$E = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 0 \end{pmatrix}, \qquad F = \begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & 0 \end{pmatrix}.$$

Then

$$\begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 1 \end{pmatrix} = EF \approx FE = \mathbf{T}.$$

Williams' Classification Theorem and Theorem 3.3 demonstrate that there exists a 2-block factor Φ from the output space of (24) to the hidden space of (23).

4. Realization problem of multi-layer cellular neural networks

This section is devoted to extending Theorems 2.3 and 3.3 to general MCNNs. A one-dimensional MCNN is realized as

$$\frac{dx_i^{(l)}}{dt} = -x_i^{(l)} + \sum_{|k|\le d} a_k^{(l)} y_{i+k}^{(l)} + \sum_{|\ell|\le d} b_\ell^{(l)} u_{i+\ell}^{(l)} + z^{(l)}, \tag{25}$$

for some $d \in \mathbb{N}$, $1 \le l \le N \in \mathbb{N}$, $i \in \mathbb{Z}$, where

$$u_i^{(l)} = y_i^{(l-1)} \quad \text{for } 2 \le l \le N \quad \text{and}$$
$$y_i^{(l)} = f(x_i^{(l)}) = \frac{1}{2}(|x_i^{(l)} + 1| - |x_i^{(l)} - 1|).$$

First, focusing on the mosaic solution of (25), namely, $|x_i^{(l)}| > 1$ for all $i \in \mathbb{Z}$, $1 \le l \le N$, the following procedure divides the parameter space into finitely equivalent regions so that any two sets of parameters possess the same output patterns if and only if they belong in the same region. For $1 \le l \le N$, the output $y_i^{(l)} = 1$ if and only if

$$a_0^{(l)} - 1 + z^{(l)} > -\left(\sum_{0<|k|\le d} a_k^{(l)} y_{i+k}^{(l)} + \sum_{|\ell|\le d} b_\ell^{(l)} u_{i+\ell}^{(l)}\right);$$

the output $y_i^{(l)} = -1$ if and only if

$$a_0^{(l)} - 1 - z^{(l)} > \sum_{0<|k|\le d} a_k^{(l)} y_{i+k}^{(l)} + \sum_{|\ell|\le d} b_\ell^{(l)} u_{i+\ell}^{(l)}.$$

Recall that

$$V^n = \{v \in \mathbb{R}^n : v = (v_1, v_2, \dots, v_n), \text{ and } |v_i| = 1, 1 \le i \le n\},$$

where $n = 4d + 1$. Let $\alpha^{(l)} = (a_{-d}^{(l)}, \dots, a_{-1}^{(l)}, a_1^{(l)}, \dots, a_d^{(l)})$, $\beta^{(l)} = (b_{-d}^{(l)}, \dots, b_d^{(l)})$. Similarly, as discussed in the previous section, the collection of admissible local pattern centered by "+1" and "−1" are

$$\mathcal{B}_+^{(l)} = \{v \diamond w : (v, w) \in V^n, a^{(l)} - 1 + z^{(l)} > -(\alpha^{(l)} \cdot v + \beta^{(l)} \cdot w)\}$$

and

$$\mathcal{B}_-^{(l)} = \{v \diamond w : (v, w) \in V^n, a^{(l)} - 1 - z^{(l)} > \alpha^{(l)} \cdot v + \beta^{(l)} \cdot w\},$$

respectively. The basic set of admissible local patterns for the lth layer of (25) is then recorded as $\mathcal{B}^{(l)} = (\widetilde{\mathcal{B}}_+^{(l)}, \widetilde{\mathcal{B}}_-^{(l)})$, where

$$\widetilde{\mathcal{B}}_+^{(l)} = \{v_+ \diamond w : v \diamond w \in \mathcal{B}_+^{(l)}\},$$
$$\widetilde{\mathcal{B}}_-^{(l)} = \{v_- \diamond w : v \diamond w \in \mathcal{B}_-^{(l)}\},$$

and $v_+ \in V^{2d+1}$ (resp. $v_- \in V^{2d+1}$) is obtained by inserting 1 (resp. −1) at the center coordinate of $v \in V^{2d}$. Denote the parameters of (25) by $(\mathbb{A}, \mathbb{B}, \mathbb{z})$, herein

$$\mathbb{A} = (A^{(1)}, \dots, A^{(N)}), \qquad \mathbb{B} = (B^{(1)}, \dots, B^{(N)}), \quad \text{and}$$
$$\mathbb{z} = (z^{(1)}, \dots, z^{(N)}).$$

Then the admissible local patterns induced by $(\mathbb{A}, \mathbb{B}, \mathbb{z})$ can be denoted by

$$\mathcal{B}(\mathbb{A}, \mathbb{B}, \mathbb{z}) = (\mathcal{B}^{(1)}, \mathcal{B}^{(2)}, \dots, \mathcal{B}^{(N)}).$$

Theorem 4.1 reveals the necessary and sufficient conditions for the existence of admissible local patterns of (25) and indicates that the parameter space $\mathscr{P} = \mathbb{R}^{N(n+2)}$ is divided into finitely equivalent regions. The proof of Theorem 4.1 is similar to the demonstration of Theorem 2.1 with a minor modification and can be routinely verified; thus, specific details are omitted here.

Theorem 4.1. *A collection of local patterns $\mathcal{B}$ is the basic set of admissible local patterns of some MCNNs if and only if $\mathcal{B} = (\mathcal{B}^{(1)}, \mathcal{B}^{(2)}, \dots, \mathcal{B}^{(N)})$ such that $\mathcal{B}^{(l)}$ satisfies Theorem 2.3 for $1 \le l \le N$. Furthermore, there exists $K \in \mathbb{N}$ and a unique collection of open sets $\{P_k\}_{k=1}^K$ of $\mathscr{P}$ satisfying*

(i) $\mathscr{P} = \cup_{k=1}^K \overline{P}_k$.
(ii) $P_k \cap P_\ell = \varnothing$ *for all* $k \ne \ell$.
(iii) $\mathcal{B}(\mathbb{A}, \mathbb{B}, \mathbb{z}) = \mathcal{B}(\mathbb{A}', \mathbb{B}', \mathbb{z}')$ *if and only if* $(\mathbb{A}, \mathbb{B}, \mathbb{z}), (\mathbb{A}', \mathbb{B}', \mathbb{z}') \in P_k$ *for some k.*

The rest of this section concentrates on the realization problem of multi-layer cellular neural networks. The scheme for investigating the realization problem of MCNNs is similar to the discussion in the previous section with more complicated calculations.

Substitute the output patterns "−1" and "+1" by − and +, respectively. First, assign each pattern in $\{-,+\}^{D\times N}$ in order so that the ordering matrix $\mathbb{X}_{D\times N}$ is well-defined, where $D = 2d+1$ is the width of admissible local patterns. Define $\chi : \{-,+\} \to \{0, 1\}$ and $\eta : \{-,+\}^{k\times l} \to \mathbb{N}$ as

$$\chi(-) = 0, \qquad \chi(+) = 1,$$

and

$$\eta\left(x = \begin{matrix} x_{1,l} & x_{2,l} & \cdots & x_{k,l} \\ x_{1,l-1} & x_{2,l-1} & \cdots & x_{k,l-1} \\ \vdots & \vdots & \ddots & \vdots \\ x_{1,1} & x_{2,1} & \cdots & x_{k,1} \end{matrix}\right) = 1 + \sum_{i,j} \chi(x_{i,j}) \cdot 2^{l(k-i)+(l-j)},$$

respectively. For example, the patterns in the collection $\{-,+\}^{2\times 2}$ are ordered as

$$\eta\begin{pmatrix} -- \\ -- \end{pmatrix} = 1, \qquad \eta\begin{pmatrix} -+ \\ -- \end{pmatrix} = 2, \dots, \eta\begin{pmatrix} +- \\ ++ \end{pmatrix} = 15,$$
$$\eta\begin{pmatrix} ++ \\ ++ \end{pmatrix} = 16.$$

The ordering matrix $\mathbb{X}_{D\times N}$ of (25) is then well-defined according to the η-ordered set $\{-,+\}^{(D-1)\times N}$. Once the basic set of admissible local patterns $\mathcal{B}$ is obtained, the transition matrix $\mathbf{T} \equiv \mathbf{T}(\mathcal{B}) \in \mathcal{M}_{(D-1)N}(\{0, 1\})$ is defined as

$$\mathbf{T}(p, q) = 1 \quad \text{if and only if } \mathbb{X}(p, q) \in \mathcal{B},$$

where $1 \le p, q \le 2^{(D-1)N}$. Let $\mathcal{A}_D = \{-,+\}^{D\times 1} = \{\alpha_i\}_{i=1}^{2^D}$ be an alphabet consisting of binary patterns of dimension $D \times 1$, where α_i presents the symbol $x = x_1 \dots x_D \in \{-,+\}^{D\times 1}$ satisfying $i = 1 + \Sigma_{j=1}^D \chi(x_j) 2^{D-j}$. Express $\mathbb{X}(p, q)$, $1 \le p, q \le 2^{(D-1)N}$, as $\alpha_{k_N} \diamond \alpha_{k_{N-1}} \diamond \cdots \diamond \alpha_{k_1}$. For $1 \le \ell \le N$, let $\mathbf{S}^{(\ell)} \equiv \mathbf{S}^{(\ell)}(\mathcal{B})$ be the ℓth symbolic transition matrix over $\mathcal{A}_D \cup \{\varnothing\}$ and be defined as

$$\mathbf{S}^{(\ell)}(p, q) = \begin{cases} \alpha_{k_\ell}, & \text{if } \mathbf{T}(p, q) = 1, \mathbb{X}(p, q) = \alpha_{k_N} \diamond \cdots \diamond \alpha_{k_1}; \\ \varnothing, & \text{otherwise.} \end{cases} \tag{26}$$

Let

$$\mathbf{Y}^{(\ell)} = \{(y_i^{(\ell)})_{i\in\mathbb{Z}} : (y_i^{(N)} \diamond y_i^{(N-1)} \diamond \cdots \diamond y_i^{(1)})_{i\in\mathbb{Z}} \in \mathbf{Y}\}$$

for $1 \le \ell \le N$. It is demonstrated that $\mathbf{S}^{(\ell)}$ well describes the topological structure of the ℓth hidden space $\mathbf{Y}^{(\ell)}$, and $\mathbf{S}^{(N)}$ describes the topological structure of the output space $\mathbf{Y}^{(N)}$ completely. The reader is referred to Ban and Chang (2013) and Ban et al. (2009) for more details.

We denote $\mathbf{S}$ as $\mathbf{S}^{(N)}$ to avoid ambiguity. The realization problem of (25) starts from partitioning $\mathbf{S}$ into an ordered set of symbolic matrices $\{\mathbf{S}_k\}_{k=1}^{2^D}$ as follows. For $1 \le k \le 2^D$, define $\mathbf{S}_k$ as the projection of $\mathbf{S}$ that only records the symbol α_k. More precisely,

$$\mathbf{S}_k(p, q) = \begin{cases} \alpha_k, & \mathbf{S}(p, q) = \alpha_k; \\ \varnothing, & \text{otherwise.} \end{cases} \tag{27}$$

$\mathbf{S} = \Sigma_{k=1}^{2^D}\mathbf{S}_k$ are derived immediately. Let $\mathbf{T}_k$ be the incidence matrix of $\mathbf{S}_k$, i.e.,

$$\mathbf{T}_k(p,q) = \begin{cases} 1, & \mathbf{S}_k(p,q) \neq \varnothing; \\ 0, & \text{otherwise.} \end{cases}$$

Denote $\mathbf{T}_k^*$ as the matrix over the trivial Boolean algebra $\mathscr{B} = \{0, e\}$ which is obtained by replacing 1's in $\mathbf{T}_k$ by e's. Suppose $\mathbb{G}$ is the free semigroup with generators $\{\mathbf{T}_k^*\}_{k=1}^{2^D}$. For $1 \leq k \leq 2^D$, let $\mathbf{A}_k$ be a 0–1 matrix indexed by $(\mathbb{G} \setminus \{\mathbf{0}\}) \times \{1, 2, \ldots, 2^D\}$ defined as

$$\mathbf{A}_k((R,i),(S,j)) = \begin{cases} 1, & i = k, R = \mathbf{T}_k^* S; \\ 0, & \text{otherwise.} \end{cases} \tag{28}$$

Furthermore, let $\mathbf{A}$ be the matrix obtained by deleting the ith row and the ith column of $\Sigma\mathbf{A}_k$ if either of which is a zero vector. Theorem 4.2 generalizes Theorem 3.3 in multiple layers cases.

Theorem 4.2. *Suppose* $\mathbf{T}$ *and* $\mathbf{S}$ *are the transition matrix and symbolic transition matrix of the output space* $\mathbf{Y}^{(N)}$ *of (25), respectively. Let* $\mathbf{A}$ *be defined as above. Then there is a single layer cellular neural network with output space* $\mathbf{Y}'$ *and a finite-to-one factor* $\Phi : \mathbf{Y}' \to \mathbf{Y}^{(N)}$ *if and only if* $\mathbf{A} \approx \mathbf{U}$*, where* $\mathbf{U}$ *is the transition matrix of* $\mathbf{Y}'$.

Proof. Observe that $\mathscr{B}$ is trivial Boolean algebra and $\mathbb{G}$ is a free semigroup generated by $\{\mathbf{T}_k^*\}_{k=1}^{2^D}$ inferring the following:

(a) $(M_1M_2)^* = M_1^*M_2^*$.
(b) $M_{i_1}\cdots M_{i_k} = 0$ if and only if $M_{i_1}^*\cdots M_{i_k}^* = 0$.
(c) $\mathbb{G}$ is finite.

Let $\phi : \mathbb{G} \times \mathbb{G} \to \mathbb{G}$ be defined as

$$\phi(M_1, M_2) = \begin{cases} M_1, & \text{if } \mathbf{A}((M_1,i),(M_2,j)) = 1 \text{ for some } i, j; \\ 0, & \text{otherwise.} \end{cases}$$

Assume $\mathbf{X}$ is the space represented by $\mathbf{A}$. The construction of $\mathbf{A}$ reveals that $\overline{\Phi} : \mathbf{X} \to \mathbf{Y}^{(N)}$ is a finite-to-one factor since $\mathbb{G}$ is finite, where $\overline{\Phi}$ is obtained from ϕ by shifting coordinates.

Let $\mathbf{Y}'$ be the output space of some single layer cellular neural network and $\mathbf{U}$ is the transition matrix of $\mathbf{Y}'$. Williams' Classification Theorem demonstrates that $\mathbf{X} \cong \mathbf{Y}'$ if and only if $\mathbf{A} \approx \mathbf{U}$. For the case that $\mathbf{A} \approx \mathbf{U}$, there exists a one-to-one and onto map $\Psi : \mathbf{Y}' \to \mathbf{X}$. The map $\Phi = \overline{\Phi} \circ \Psi$ is the desired conjugacy between $\mathbf{Y}'$ and $\mathbf{Y}^{(N)}$. This completes the proof. □

5. Conclusion and further discussion

This paper investigates whether the output space of a multi-layer cellular neural network can be realized via a single layer cellular neural network based on the existence of finite-to-one maps from one output space to another. Theorem 4.2 reveals a necessary and sufficient condition for the realization problem of multi-layer cellular neural networks. Notably, the proposed methodology can be applied to realizing the specific hidden space of a multi-layer cellular neural network via a single layer system. We summarize the procedure of the general cases of Theorem 4.2 as follows.

Assume $\mathbf{Y}$ is the mosaic solution space of a multi-layer cellular neural network with d-nearest neighborhood, and $\mathbf{T}$ is its transition matrix. Let $D = 2d + 1$ and $\mathcal{S} = \{\alpha_i\}_{i=1}^{D}$, where

$$\alpha_1 = --\cdots--, \qquad \alpha_2 = --\cdots-+,$$
$$\alpha_3 = --\cdots+-, \ldots, \alpha_D = ++\cdots++.$$

Step 1 Construct the symbolic transition matrix $\mathbf{S}^{(\ell)}$ from $\mathbf{T}$ as defined in (26) for the ℓth hidden space $\mathbf{Y}^{(\ell)}$. (The output space $\mathbf{Y}^{(N)}$ is considered as a special case where $\ell = N$.) Extract $\{\mathbf{S}_k\}_{k=1}^{2^D}$ from $\mathbf{S}^{(\ell)}$ as defined in (27).

Step 2 Let $\mathbb{G}$ be the semigroup generated by the matrices, which come from $\mathbf{S}_k$ for $1 \leq k \leq 2^D$, over trivial Boolean algebra, and let $\mathbf{A}$ be the matrix obtained by deleting both the ith row and the ith column of $\Sigma\mathbf{A}_k$ if either one of them is zero, where $\{\mathbf{A}_k\}_{k=1}^{2^D}$ as in (28). Then there is a finite-to-one factor map $\overline{\Phi}$ from $\mathbf{X}$ to $\mathbf{Y}^{(\ell)}$, herein $\mathbf{X}$ is the space whose transition matrix is $\mathbf{A}$.

Step 3 Determine whether there is a single layer cellular neural network whose output space is $\mathbf{Y}'$ with transition matrix $\mathbf{U}$ such that $\mathbf{A} \approx \mathbf{U}$, i.e., $\mathbf{A}$ is strongly shift equivalent to $\mathbf{U}$. The possible selection of $\mathbf{U}$ is constrained by Theorem 2.3 and the radius d of the neighborhood.

Step 4 The strong shift equivalence of $\mathbf{A}$ and $\mathbf{U}$ is a necessary and sufficient condition for the existence of conjugacy Ψ : $\mathbf{Y}' \to \mathbf{X}$. The desired finite-to-one factor $\Phi : \mathbf{Y}' \to \mathbf{Y}^{(\ell)}$ for realizing the ℓth hidden space (or the output space) of a multi-layer cellular neural network via the output space of a single layer cellular neural network is then defined as $\Phi = \overline{\Phi} \circ \Psi$.

$$\begin{tikzcd} \mathbf{Y}' \arrow[r, "\Psi"] \arrow[dr, "\Phi"'] & \mathbf{X} \arrow[d, "\overline{\Phi}"] \\ & \mathbf{Y} \end{tikzcd}$$

This realization problem naturally leads to a question: when does the finite-to-one map, if it exists, become one-to-one? Notably, the existence of Φ depends on the existence of Ψ. Since $\Phi = \overline{\Phi} \circ \Psi$ and Ψ, if it exists, is always a conjugacy, this is same as asking when does $\overline{\Phi}$ become one-to-one. Future studies could further discuss this topic.

Other interesting topics related to this remain to be discussed. To conclude this study, we point out some of the biological relevance of neural networks. Countless studies have investigated the application of multi-layer and single layer cellular neural networks in biology. As follows, we offer brief illustration of some of these works.

Neural networks were developed initially to model biological functions such as human brains or visual systems. They are intelligent, thinking machines, and they learn from experience in a way that no conventional computer can. Moreover, they can rapidly solve hard computational problems. The adaptive resonance theory and neocognitron were inspired by the organization of the visual nervous system.

The neocognitron is a neural network model proposed by Fukushima, based on neurophysiological findings drawn from the visual systems of mammals. There are two major types of cells in the neocognitron, and the neocognitron consists of the cascaded connection of a number of modules, each of which consists of a layer of one type of cell followed by a layer of another type of cell. The powerful capabilities and computational complexity of the neocognitron are acquired from the architecture of the cascaded connection of mixed layers of different types of cells in a hierarchical network. This architecture is useful for recognizing robust visual patterns and image processing.

Although the model of the neocognitron is different from that of a cellular neural network, the methodology proposed for the realization problem in multi-layer cellular neural networks can be extended to the case of the neocognitron. Related works are in the preparation phase. For further references related to the application of multi-layer networks in biology, refer to Fukushima (2013a, 2013b), Lin, Simossis, Taylor, and Heringa (2005), Mattick (2003), Tarca, Carey, Chen, Romero, and Drăghici (2007) and the references therein.

Acknowledgments

The authors wish to express their gratitude to the anonymous referees for their careful reading and useful suggestions, which make significant improvements to this work. Further elucidation related to the present paper is also inspired by their comments and is in the preparation phase. This work is partially supported by the National Science Council, ROC (Contract Nos. NSC 102-2628-M-259-001-MY3 and 103-2115-M-390-004-).

References

Arena, P., Baglio, S., Fortuna, L., & Manganaro, G. (1998). Self-organization in a two-layer CNN. *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, *45*, 157–162.
Ban, J.-C., & Chang, C.-H. (2009). On the monotonicity of entropy for multi-layer cellular neural networks. *International Journal of Bifurcation and Chaos in Applied Sciences and Engineering*, *19*, 3657–3670.
Ban, J.-C., & Chang, C.-H. (2013). The learning problem of multi-layer neural networks. *Neural Networks*, *46*, 116–123.
Ban, J.-C., Chang, C.-H., Lin, S.-S., & Lin, Y.-H. (2009). Spatial complexity in multi-layer cellular neural networks. *Journal of Differential Equations*, *246*, 552–580.
Bengio, Y. (2009). Learning deep architectures for AI. *Foundations and Trends in Machine Learning*, *2*, 1–127.
Bengio, Y., & LeCun, Y. (2007). Scaling learning algorithms towards AI. In L. Bottou, O. Chapelle, D. DeCoste, & J. Weston (Eds.), *Large scale kernel machines*. MIT Press.
Carmona, R., Jimenez-Garrido, F., Dominguez-Castro, R., Espejo, S., & Rodriguez-Vazquez, A. (2002). CMOS realization of a 2-layer cnn universal machine chip. In *Proceedings of the 2002 7th IEEE international workshop on cellular neural networks and their applications, 2002. CNNA 2002* (pp. 444–451).
Chang, C.-H. (2015). Deep and shallow architecture of multi-layer neural networks. *IEEE Transactions on Neural Networks and Learning Systems*, http://dx.doi.org/10.1109/TNNLS.2014.2387439 (published online).
Chua, L. O. (1998). *World Scientific Series on Nonlinear Science, Series A*: *vol. 31*. *CNN: a paradigm for complexity*. Singapore: World Scietific.
Chua, L. O., & Roska, T. (2002). *Cellular neural networks and visual computing*. Cambridge University Press.
Chua, L. O., & Shi, B. E. (1991). Multiple layer cellular neural networks: A tutorial. In E. F. Deprettere, & A.-J. van der Veen (Eds.), *Algorithms and parallel VLSI architectures* (pp. 137–168). Amsterdam, The Netherlands: Elsevier.
Chua, L. O., & Yang, L. (1988a). Cellular neural networks: Applications. *IEEE Transactions on Circuits and Systems*, *35*, 1273–1290.
Chua, L. O., & Yang, L. (1988b). Cellular neural networks: Theory. *IEEE Transactions on Circuits and Systems*, *35*, 1257–1272.
Coven, E. M., & Paul, M. (1975). Sofic systems. *Israel Journal of Mathematics*, *20*, 165–177.
Crounse, K. R., & Chua, L. O. (1995). Methods for image processing and pattern formation in cellular neural networks: A tutorial. *IEEE Transactions on Circuits and Systems*, *42*, 583–601.
Crounse, K. R., Roska, T., & Chua, L. O. (1993). Image halftoning with cellular neural networks. *IEEE Transactions on Circuits and Systems*, *40*, 267–283.
Fukushima, K. (2013a). Artificial vision by multi-layered neural networks: Neocognitron and its advances. *Neural Networks*, *37*, 103–119.
Fukushima, K. (2013b). Training multi-layered neural network neocognitron. *Neural Networks*, *40*, 18–31.
Hinton, G. E., Osindero, S., & Teh, Y. (2006). A fast learning algorithm for deep belief nets. *Neural Computation*, *18*, 1527–1554.
Hsu, C.-H., Juang, J., Lin, S.-S., & Lin, W.-W. (2000). Cellular neural networks: Local patterns for general template. *International Journal of Bifurcation and Chaos in Applied Sciences and Engineering*, *10*, 1645–1659.
Li, X. (2009). Analysis of complete stability for discrete-time cellular neural networks with piecewise linear output functions. *Neural Computation*, *21*, 1434–1458.
Lin, K., Simossis, V., Taylor, W., & Heringa, J. (2005). A simple and fast secondary structure prediction method using hidden neural networks. *Bioinformatics*, *21*, 152–159.
Mattick, J. (2003). Challenging the dogma: the hidden layer of non-protein-coding rnas in complex organisms. *Bioessays*, *25*, 930–939.
Murugesh, V. (2010). Image processing applications via time-multiplexing cellular neural network simulator with numerical integration algorithms. *International Journal of Computer Mathematics*, *87*, 840–848.
Peng, Jun, Zhang, Du, & Liao, Xiaofeng (2009). A digital image encryption algorithm based on hyper-chaotic cellular neural network. *Fundamenta Informaticae*, *90*, 269–282.
Tarca, A., Carey, V., Chen, X.-W., Romero, R., & Drăghici, S. (2007). Machine learning and its applications to biology. *PLOS Computational Biology*, *3*, 953–963.
Török, L., & Roska, T. (2004). Stability of multi-layer cellular neural/nonlinear networks. *International Journal of Bifurcation and Chaos in Applied Sciences and Engineering*, *14*, 3567–3586.
Utgoff, P. E., & Stracuzzi, D. J. (2002). Many-layered learning. *Neural Computation*, *14*, 2497–2539.
Williams, R. F. (1973). Classification of subshifts of finite type. *Annals of Mathematics*, *98*, 120–153. Errata, Annals of Math. 99 (1974), 380–381.
Xavier-de Souza, S., Yalcin, M. E., Suykens, J. A. K., & Vandewalle, J. (2004). Toward CNN chip-specific robustness. *IEEE Transactions on Circuits and Systems. I. Regular Papers*, *51*, 892–902.
Yang, Z., Nishio, Y., & Ushida, A. (2001). A two layer CNN in image processing applications. In *Proc. of the 2001 international symposium on nonlinear theory and its applications* (pp. 67–70).
Yang, Z., Nishio, Y., & Ushida, A. (2002). Image processing of two-layer CNNs—applications and their stability. *IEICE Transactions on Fundamentals*, *E85-A*, 2052–2060.