Patterns of Synchrony for Feed-forward and Auto-regulation Feed-forward Neural Networks

Manuela A. D. Aguiar[1], Ana Paula S. Dias[2] and Flora Ferreira[3]

[1] *Faculdade de Economia, Centro de Matemática, Universidade do Porto, Rua Dr Roberto Frias, 4200-464 Porto, Portugal; maguiar@fep.up.pt*

[2] *Dep. Matemática, Centro de Matemática, Universidade do Porto, Rua do Campo Alegre, 687, 4169-007 Porto, Portugal; apdias@fc.up.pt*

[3] *Centro de Matemática, Universidade do Porto, Rua do Campo Alegre, 687, 4169-007 Porto, Portugal; flora.ferreira@gmail.com*

November 11, 2016

Abstract

We consider feed-forward and auto-regulation feed-forward neural (weighted) coupled cell networks. In feed-forward neural networks, cells are arranged in layers such that the cells of the first layer have empty input set and cells of each other layer receive only inputs from cells of the previous layer. An auto-regulation feed-forward neural coupled cell network is a feed-forward neural network where additionaly some cells of the first layer have auto-regulation, that is, they have a self-loop. Given a network structure, a robust pattern of synchrony is a space defined in terms of equalities of cell coordinates that is flow-invariant for any coupled cell system (with additive input structure) associated with the network. In this paper we describe the robust patterns of synchrony for feed-forward and auto-regulation feed-forward neural networks. Regarding feed-forward neural networks, we show that only cells in the same layer can synchronize. On the other hand, in the presence of auto-regulation, we prove that cells in different layers can synchronize in a robust way and we give a characterization of the possible patterns of synchrony that can occur for auto-regulation feed-forward neural networks.

AMS classification scheme numbers: 34C15 37C10 06B23 05C50

Many real-world applications can be modelled by coupled cell networks which abstract the cells and the type of interaction between pairs of cells. One of the advantages of taking the network structure into account is that the network encodes information that impacts the dynamics, independently of the specific equations used to model the original application. An example is the existence of (robust) patterns of synchrony that occur for any system that has structure consistent with a given network - these are usually called the network synchrony subspaces. These patterns, that are defined in terms of equalities of cell coordinates, can be described using only the network structure. In a feed-forward neural network, cells are disposed in layers and it is possible to order the layers such that cells in one layer receive inputs only from cells from the previous layer, except for the cells of the first layer. If some cells of the first layer have self-loops we say that the cells are auto-regulated. Feed-forward and auto-regulation feed-forward neural networks are broadly used in the modelling of real-world applications in many different areas with special emphasis in neurosciense. In particular, the models that are most used to explain how the brain processes information are the feed-foward artificial neural networks. We consider both network types and we describe their robust patterns of synchrony. For feed-forward neural networks we prove that cells can only synchronize if they are in the same layer. For feed-forward neural networks with auto-regulation we show that cells in different layers can synchronize in a robust way. Moreover, for any path starting at a cell of the first layer, the synchrony pattern is characterized by the first cells being synchronized and the subsequent cells being desynchronized. In particular, the cells in the path can be all synchronized or desynchronized.

1 Introduction

Feed-forward Neural Networks are used in many practical applications in different fields such as, for example, Neuroscience [22, 25], Neural and Biomedical Engineering [5, 28, 4], Robotics [6] and Economics [26].

A *Feed-forward Neural Network* (FFNN) is a network without self-loops (auto-regulations) or cycles (feedback loops). In this type of network the cells are arranged in layers, where the information moves in only one direction, forward, from the *input nodes* (first layer), through the *hidden nodes* (middle layer(s)), and to the *output nodes* (last layer). There is no connection among cells in the same layer. Each cell in a layer only receives connections from cells in the previous layer. The sum of the weights of the connections directed to a cell is the *valency* of that cell. Thus, in a FFNN, the cells of the first layer have valency zero. According to the concept of artificial neural network, cells in the same layer must be of the same type. Here we consider feed-forward neural networks where all cells are of the same type from the dynamics point of view. That is, all the cells of the network have the same (internal) dynamics if the connections between the cells are switched off.

In this paper, as well as FFNNs, we consider another special type of networks, *Auto-regulation Feed-forward Neural networks* (AFFNNs) that are FFNNs with auto-regulation input, that is, with self-loops. The AFFNNs are examples of recurrent networks.

We focus our analysis on synchronization patterns. The experimental and also theoretical study of synchrony in FFNNs and AFFNNs and its impact, from the applications point of view, has attracted the interest of many scientists in the last decades. Different studies have come to the conclusion that synchrony in such networks seems to be the explanation for various phenomena as, for example, the precisely timed spike patterns of the brain observed in experiments. For instance, there is evidence that the brain exhibits synchronous firing patterns in its normal functioning, which may be crucial for information processing, but also synchronous pathological events as those occurring, for example, during a seizure. See [7], [11], [17], [21], [23], [29] and references therein.

Our approach is from the point of view of the theory of coupled cell networks developed by Golubitsky, Stewart and collaborators [27, 15] and Field [9]. In the context of this theory, Aguiar and Dias [1] give a characterisation, in terms of the eigenvectors of the adjacency matrices of a network, of the patterns of synchrony of the network and provide an algorithm to compute those patterns. Although they consider networks with nonnegative integer adjacency matrices their results follow trivially for the more general situation that we are considering here of networks with real adjacency matrices. In this work, since the networks have a special structure, a feed-forward structure, we are able to give a more specific characterisation of the patterns of synchrony for this kind of networks explicitly in terms of the topological structure of the network, and to give a simpler algorithm to find those synchrony patterns. We start by defining an extension of the aforementioned theory to accommodate the fact that in the types of networks that we are considering, the most commonplace, is that each edge has an associated numerical value called a weight, which in principle, is not necessarily a nonnegative integer number. Note that in the theory of coupled cell networks, as self-loops and multiarrows (arrows with the same head and tail cells) are allowed, this corresponds to take classes of networks where the connections weights are all nonnegative integer numbers. We define coupled cell systems in a way that codifies general weights for connections by considering coupled cell systems with additive input structure. We then remark that the results of [27, 15] concerning the characterization of the network synchrony patterns are valid as well in our setup where the proofs are a trivial extension (or restriction) of the proofs presented in [27, 15].

We focus then at the possible patterns of robust synchrony that can occur for FFNNs and AFFNNs. Our interest lies, not at a particular network, but at intrinsic properties for each class of networks with respect to synchronisation. Moreover, we provide an insight at genuine and relevant differences between these two types of networks corresponding to their performances at the synchrony patterns that can occur due to their distinct architecture types. We prove that, in the patterns of synchrony that can occur for a given FFNN structure each group of cells behaving synchronously is contained in a unique layer. That is, there cannot be synchronous cells in different layers. This fact is implicit in one of the questions that the neuroscience community has been devoted to trying to understand, that is how synchronous activity may propagate along the layers of a FFNN, Diesmann *et al.* [7], Jahnke *et al.*[17]. We characterise the set of all patterns of synchrony of a FFNN based only on its connectivity structure. Taking into account the conclusion in Nowotny and Huerta [21] that synchrony in feed-forward neural networks is independent of the neuron internal dynamics and results entirely from the network topology, it follows that the patterns of synchrony that we obtain for a given network structure constitute the complete set of patterns of synchrony for any neural dynamics with that network topology.

In contrast to FFNNs, for AFFNNs, cells in different layers can synchronize in a robust way. Given a cell, we can consider the cell *input subnetwork* given by the subnetwork formed by all paths (and the cells involved in the paths) directed to the cell (Definition 4.3). We prove that, for AFFNNs, if the input subnetwork of a cell contains another cell synchronized with it, then all the cells of the input subnetwork are synchronized (with the cell).

The paper is organized in the following way. In Section 2 we introduce briefly the basics on coupled cell networks and synchronization considering our extension to the coupled cell network formalism of Golubitsky and Stewart. The formal definition of FFNNs is given in Section 3 and the main result on synchronization in FFNNs is given by Theorem 3.4. In Section 4 we describe AFFNNs and we characterize their robust patterns of synchrony. Our main result is Theorem 4.8. In both sections 3 and 4 we propose a simple algorithm to enumerate the robust patterns of synchrony (Algorithms 3.10 and 4.13). The algorithms are implemented in Python and available from `http://www.fc.up.pt/cmup/adfsoftware`.

2 Background

Given a network structure G, that is, a weighted directed graph, a coupled cell system consistent with G is a network of interacting individual dynamical systems − the cells. Thus the nodes of the graph represent the cells and the arrows of the graph the interactions or couplings. Following [27, 15, 9], we take a cell to be a system of ordinary differential equations.

Let $\mathcal{C} = \{1, ..., n\}$ denote the set of cells of the network. Each coupled cell c in $\mathcal{C}$ is associated with a phase space P_c, which is assumed to be a nonzero finite-dimensional real vector space, say $\mathbf{R}^k$, for some $k > 0$. If cells c and d are assumed identical, it is required then that $P_c = P_d$, that is, the two spaces must be identified canonically, and the internal dynamics of the cells is defined by the same differential equation.

We consider *weighted networks* of equivalent cells where there is only one kind of coupling which can have associated a different weight. A system associated with cell j of such an n-cell weighted network G has the form

$$\dot{x}_j = f(x_j) + \sum_{i=1}^{n} w_{ji} g\left(x_j, x_i\right),\ j = 1, ...n, \tag{2.1}$$

where $f : \mathbf{R}^k \to \mathbf{R}^k$ and $g : \mathbf{R}^k \times \mathbf{R}^k \to \mathbf{R}^k$ are smooth functions; also, each $w_{ji} \in \mathbf{R}$ is the value of the weight of the coupling strength from cell i to cell j. In particular, the equality $w_{ji} = 0$ occurs when there is no connection from cell i to cell j. Note that the function f characterizes the *internal dynamics*. Moreover, the function g is the *coupling function*. Thus, we are assuming $x_i \in \mathbf{R}^k$, for $k \geq 1$. When $k > 1$, the term $w_{ji} g\left(x_j, x_i\right)$ refers to scalar multiplication. We say that the coupled cell system (2.1) is G-*admissible* and denote by $W = [w_{ij}]_{1 \leq i,j \leq n}$ the $n \times n$ *weighted adjacency matrix* of G.

Coupled cell systems of the form (2.1) are a special class of coupled cell systems with *additive input structure*, see Definition 2.9 of Field [10], which as mentioned there, allows the addition and deletion of connections. Moreover, networks of Kuramoto phase oscillators and pulse coupled systems are coupled cell systems with additive input structure, see for example, Ashwin *et al.* [3] and Neves and Timme [19].

Example 2.1 Consider the weighted networks G (left) and Q (right) in Figure 1. The arrows correspond to directed edges between two cells with weight equal to value written above. The weight connection is 0 when there is no arrow between two cells. The network weighted adjacency matrices are

$$W = \begin{bmatrix} 1 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \end{bmatrix} \text{ and } \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 2 & 0 \end{bmatrix},$$

respectively. A coupled cell system admissible for the network G has the form:

$$\begin{array}{rcl} \dot{x}_1 & = & f\left(x_1\right) + g\left(x_1, x_1\right) - g\left(x_1, x_2\right) \\ \dot{x}_2 & = & f\left(x_2\right) \\ \dot{x}_3 & = & f\left(x_3\right) + g\left(x_3, x_1\right) \\ \dot{x}_4 & = & f\left(x_4\right) + g\left(x_4, x_2\right) \\ \dot{x}_5 & = & f\left(x_5\right) + g\left(x_5, x_3\right) + g\left(x_5, x_4\right) \end{array} \tag{2.2}$$

where $f : \mathbf{R}^k \to \mathbf{R}^k$ and $g : \mathbf{R}^k \times \mathbf{R}^k \to \mathbf{R}^k$ are smooth functions. We are assuming that the internal cell phase space is $\mathbf{R}^k$ and so the total phase space $\mathrm{P} = \left(\mathbf{R}^k\right)^5$. Note that equations (2.2) restricted to the polydiagonal subspace Δ defined by $x_1 = x_2$, $x_3 = x_4$, that is when cells 1 and 2 are synchronized and cells 3 and 4 are synchronized, are

$$\begin{array}{l} \dot{x}_2 = \dot{x}_1 = f(x_1) \\ \dot{x}_4 = \dot{x}_3 = f(x_3) + g\left(x_3, x_1\right) \\ \dot{x}_5 = f(x_5) + 2g\left(x_5, x_3\right) \end{array},$$

and are consistent with the network Q. We see below that the network Q is the quotient network of the network G by the (balanced) relation on the network set of cells with classes $\{1, 2\}$, $\{3, 4\}$, $\{5\}$. ◇

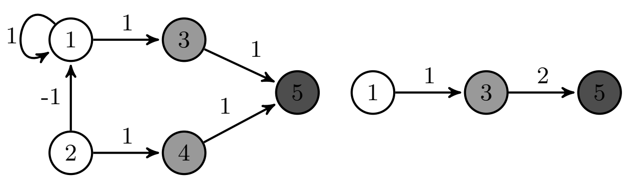

Figure 1: (Left) A 5-cell weighted network G. (Right) A 3-cell weighted network Q. Any coupled cell system consistent with G restricted to the polydiagonal subspace defined by $x_1 = x_2$, $x_3 = x_4$, is consistent with Q. That is, the network Q is the quotient network of the network G by the (balanced) relation with classes $\{1, 2\}$, $\{3, 4\}$, $\{5\}$.

2.1 Patterns of synchrony

In [27, 15] the concept of network synchrony pattern is defined. More precisely, given an n-cell network structure G, a subspace Δ of P defined by certain equalities of coordinates is said to be a *network synchrony subspace* when it is left invariant under the flow of any G-admissible coupled cell system. Thus, for any system $\dot{X} = F(X)$, where F is G-admissible, $F\left(\Delta\right) \subseteq \Delta$.

In [27, 15] a necessary and sufficient condition is also given in terms of the network structure, for such a subspace Δ (which is also called a *polydiagonal space*) to be a synchrony subspace. Precisely, consider the equivalence relation $\bowtie$ on the network set of cells defined in the following way: $i \bowtie j$ if and only if all the vectors in Δ satisfy the equality $x_i = x_j$. Write then $\Delta = \Delta_{\bowtie}$. In [27, 15] it is proved that $\Delta_{\bowtie}$ is a network synchrony space if and only if $\bowtie$ satisfies certain conditions, in which case, $\bowtie$ is said to be *balanced*. We describe now the conditions for $\bowtie$ to be balanced. Two cells i and j are said to be $\bowtie$-related for balanced $\bowtie$ when there is a bijection between the sets of directed edges to i and j which preserves the edge types and the $\bowtie$-class of the edges tail cells. Moreover, it is remarked, for example in [1], that $\bowtie$ is balanced if and only if $\Delta_{\bowtie}$ is left invariant under the network adjacency matrices.

In this section, we adapt the above definition of balanced relation to networks where the network adjacency matrix is a weighted matrix. An analogue to Definition 6.4 of [27] is the following:

Definition 2.2 An equivalence relation $\bowtie$ on the set of cells $\{1, \ldots, n\}$ of a network is said to be *balanced* when satisfies the following condition: we have $i \bowtie j$ if and only if the sum of the weights of the couplings directed to i and j, from cells in the same $\bowtie$-class, are equal. $\diamond$

Following [15], an equivalence relation $\bowtie$ can be visualized graphically by colouring equivalent cells with the same colour. Then, by Definition 2.2, $\bowtie$ is balanced if and only if, whenever two cells i and j have the same colour, the sums of the weights of the couplings directed to i and j from cells of the same colour, are equal.

Examples 2.3 (i) Returning to the network G on the left of Figure 1, the relation $\bowtie$ with classes $\{1, 2\}$, $\{3, 4\}$, $\{5\}$ is balanced: cells in $\{1, 2\}$ and the cells in $\{3, 4\}$ only receive couplings from cells in $\{1, 2\}$; the sum of the weights of the input couplings to cells 1 and 2 is 0, and to cells 3 and 4 is 1.
(ii) Consider the network in Figure 2. The equivalence relation on the network set of cells $\mathcal{C} = \{1, \ldots, 12\}$ with classes with classes $I_1 = \{1, 2, 3\}$, $I_2 = \{4, 5, 6\}$, $I_3 = \{7\}$, $I_4 = \{8\}$, $I_5 = \{9, 10\}$ and $I_6 = \{11, 12\}$ is balanced: every cell in I_1 has no inputs; every cell in I_2 has inputs from cells in I_1 with weight sum 2; every cell in I_5 has inputs from cells in I_2 with weight sum -1 and from cell 7 (with weight 2), and every cell in I_6 has an input from cell 8 (with weight -1) and from cells I_5 with weight sum 1.5. $\diamond$

If we take now admissible coupled cell systems with additive input structure as defined in (2.1), then we also have an analogue to Theorem 6.5 of [27]:

Theorem 2.4 *Let G be an n-cell weighted network. Consider the admissible coupled cell systems for G, as in (2.1), for a given choice of total phase space $\left(\mathbf{R}^k\right)^n$. Then, a polydiagonal subspace $\Delta_{\bowtie}$ is a synchrony subspace for G if and only if the $\bowtie$-relation is balanced on the set of cells of G.* $\diamond$

Proof Trivially, if $\bowtie$ is balanced, then for any coupled cell system (2.1), for a given choice of the internal phase space $\mathbf{R}^k$, if $x = (x_1, \ldots, x_n) \in \Delta_{\bowtie}$ and $i \bowtie j$, the equations for cell i and j evaluated at x coincide. Thus, $\Delta_{\bowtie}$ is flow-invariant for equations (2.1).

To prove the inverse, we assume that for any coupled cell system as (2.1) for a given choice of $\mathbf{R}^k$, $\Delta_{\bowtie}$ is flow-invariant for (2.1), and show that then $\bowtie$ has to be balanced. An analogue of the proof given in Theorem 6.5 of [27] can be given and uses the fact that $\Delta_{\bowtie}$ has to be, in particular, left invariant taking linear admissible vector fields. Briefly, let $W = [w_{ij}]$ be the weighted adjacency matrix of G, and consider the following (linear) G-admissible coupled cell system:

$$\dot{x}_j = \sum_{i=1}^{n} w_{ji} x_i, \; j = 1, \ldots, n, \tag{2.3}$$

where we are taking $x_i \in \mathbf{R}$, for $i = 1, \ldots, n$. We have that (2.3) leaves $\Delta_{\bowtie}$ invariant if and only if the weighted adjacency matrix W leaves $\Delta_{\bowtie}$ invariant. $\square$

Example 2.5 Taking G to be the network on the left of Figure 1, recall that the relation $\bowtie$ with classes $\{1, 2\}$, $\{3, 4\}$, $\{5\}$ is balanced. Thus, by Theorem 2.4, the polydiagonal $\Delta_{\bowtie}$ defined by the equalities $x_1 = x_2, x_3 = x_4$ is a network synchrony subspace. As well, for the network in Figure 2 since the equivalence relation on the network set of cells $\mathcal{C} = \{1, \ldots, 12\}$ with classes $I_1 = \{1, 2, 3\}$, $I_2 = \{4, 5, 6\}$, $I_3 = \{7\}$, $I_4 = \{8\}$, $I_5 = \{9, 10\}$ and $I_6 = \{11, 12\}$ is balanced, the polydiagonal $\Delta_{\bowtie}$ defined by the equalities $x_1 = x_2 = x_3, x_4 = x_5 = x_6, x_9 = x_{10}, x_{11} = x_{12}$ is a network synchrony subspace. $\diamond$

Definition 2.6 We call the *valency* of a cell c of a network G the sum of the weights of the couplings directed to c. $\diamond$

Remark 2.7 A necessary condition for an equivalence relation on the set of cells of a network to be balanced is that cells in the same class must have the same valency. $\diamond$

It is proved in Section 7 of [27] that if $\Delta_{\bowtie}$ is a synchrony subspace for a network G then any G-admissible coupled cell system restricted to $\Delta_{\bowtie}$ is an admissible coupled cell system for a smaller network called the *quotient network of G by $\Delta_{\bowtie}$*, and denoted by $Q = G/\bowtie$. The same holds in our setup. The network Q is obtained from G in the following way: the cells of Q correspond to the $\bowtie$-equivalence classes and the directed edges of Q are the projections of the directed edges of G. More precisely, if $\mathcal{I}_i$ and $\mathcal{I}_j$ are two $\bowtie$-equivalence classes, as $\bowtie$ is balanced,

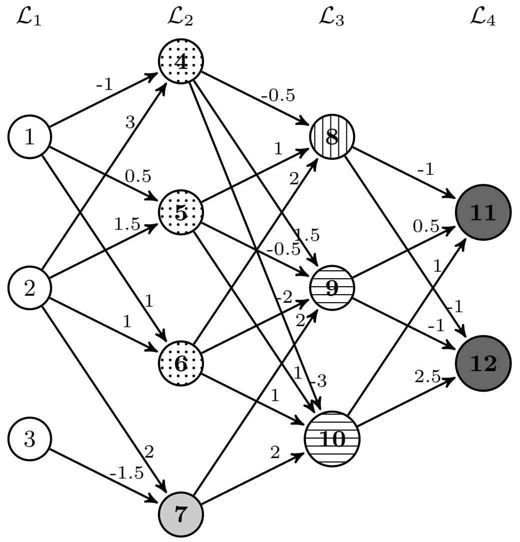

Figure 2: A FFNN with four layers and twelve cells. The equivalence relation on the network set of cells with classes $\{1, 2, 3\}$, $\{4, 5, 6\}$, $\{7\}$, $\{8\}$, $\{9, 10\}$ and $\{11, 12\}$ is balanced.

then the sum of the weights of all directed edges from the cells in $\mathcal{I}_i$ to any given cell in $\mathcal{I}_j$ does not depend on the cell of $\mathcal{I}_j$ considered.Thus if $\mathcal{I}_i = \{i_1, \ldots, i_l\}$ and we choose $j_1 \in \mathcal{I}_j$, then there will be a directed edge from $\mathcal{I}_i$ to $\mathcal{I}_j$ with weight $w_{j_1,i_1} + \cdots + w_{j_1,i_l}$. We also say that G is *a lift of* Q. In particular, it follows that the valency of each cell in Q is the valency of any cell in the lift belonging to the $\bowtie$-class of that cell.

Example 2.8 The network Q in Figure 1 (right) is the quotient of the network G in Figure 1 (left) by the $\bowtie$-equivalence relation with classes $\{1, 2\}$, $\{3, 4\}$, $\{5\}$. In fact, taking any coupled cell system admissible for G, as in (2.2), we have that, the restriction to the synchrony subspace $\Delta_{\bowtie}$ defined by $x_1 = x_2$, $x_3 = x_4$ is consistent with the network Q on the right of Figure 1. Equivalently, the network G is a lift of Q. ◇

Definition 2.9 Let G be a network and $\bowtie$ a balanced equivalence relation on the network set of cells. We say that $\bowtie$ is a *spurious balanced equivalence relation* if there is a directed edge with nonzero weight projecting in the quotient network $G/\bowtie$ into an edge with zero weight. We also say that $\Delta_{\bowtie}$ is a *spurious synchrony pattern.* ◇

See Figure 3 for two network examples where the colourings of the cells correspond to spurious balanced equivalence relations.

3 Synchronisation in FFNNs

We formalize the definition of FFNN.

Definition 3.1 Let G be an n-cell weighted network with set of cells $\mathcal{C}$. If there is a partition $\mathcal{L}_i$, $i = 1, \ldots, r$, with

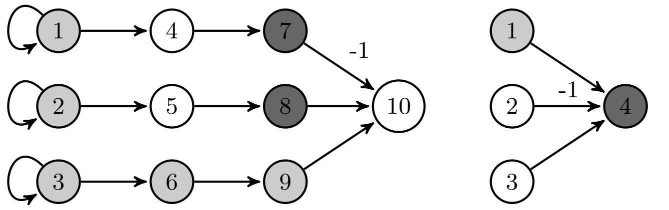

Figure 3: Two networks with cell colourings corresponding to spurious balanced equivalence relations. The edges with weight 1 have the weight omitted.

$r \leq n$, of the set $\mathcal{C}$ such that
(i) cells in $\mathcal{L}_1$ have valency zero and receive no connections;
(ii) for each $j \in \{2, ..., r\}$, all the cells in $\mathcal{L}_j$ have nonzero valency and receive connections only from cells in $\mathcal{L}_{j-1}$;
then the network G is a *FFNN* and each subset $\mathcal{L}_i$ is called a *layer.* ◇

We now characterize FFNNs using weighted adjacency matrices. According to Definition 3.1 the cells of a FFNN can be enumerated such that the corresponding weighted adjacency matrix has the following lower-triangular block form:

$$W = \begin{bmatrix} 0_{1,1} & 0_{1,2} & \ldots & 0_{1,r-1} & 0_{1,r} \\ W_{2,1} & 0_{2,2} & \ldots & 0_{2,r-1} & 0_{2,r} \\ 0_{3,1} & W_{3,2} & \ldots & 0_{3,r-1} & 0_{3,r} \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ 0_{r,1} & 0_{r,2} & \ldots & W_{r,r-1} & 0_{r,r} \end{bmatrix}. \tag{3.4}$$

Denoting by $l_i = \#\mathcal{L}_i$, for $i = 1, \ldots, r$, then the block $W_{i,i-1}$ is a $l_i \times l_{i-1}$ matrix and $0_{i,j}$ is the $l_i \times l_j$ zero matrix. The entries of $W_{i,i-1}$ correspond to the weights of the connections from the cells in layer $\mathcal{L}_{i-1}$ to the cells in layer $\mathcal{L}_i$.

Remark 3.2 Observe that, assumption (ii) in Definition 3.1 implies that the rows of each matrix $W_{i,i-1}$, for $i = 2, \ldots, r$, have nonzero sum. ◇

Example 3.3 Figure 2 shows an example of a FFNN with four layers, $\mathcal{L}_i$, $i = 1, \ldots, 4$. Note that the cells are ordered such that the weighted adjacency matrix W has lower-triangular block form as in (3.4). For example, $W_{4,3} = \begin{bmatrix} -1 & 0.5 & 1 \\ -1 & -1 & 2.5 \end{bmatrix}$. ◇

3.1 Patterns of synchrony

The next theorem states that for the synchrony patterns that can occur for a FFNN there are no two synchronous cells in different layers.

Theorem 3.4 *Cells in different layers of a FFNN cannot synchronise.*

Proof Let G be a FFNN with set of cells $\{1,\ldots,n\}$. Assume the ordering of the cells of G is such that its weighted matrix W has block structure as in (3.4). Thus $\{1,\ldots,n\} = \mathcal{L}_1 \cup \mathcal{L}_2 \cup \cdots \cup \mathcal{L}_r$, where $\mathcal{L}_i = \{l_{i-1}+1,\ldots,l_i\}$, taking $l_0 = 0$. Note that, as the valency of the cells in the layer $\mathcal{L}_1$ is zero and for the other cells is nonzero, then cells in layer $\mathcal{L}_1$ cannot synchronize with cells of other layers (Remark 2.7).

Now, assume that $\Delta_{\bowtie}$ is a synchrony subspace for G and consider the associated balanced equivalence relation $\bowtie$ on $\{1,\ldots,n\}$. Let $\mathcal{I}_1,\ldots,\mathcal{I}_k$ be the $\bowtie$-classes. Assume, by contradiction, that there is at least one $\bowtie$-class containing cells in two different layers. Let $\mathcal{L}_p$ be the first layer containing cells that synchronize with cells of other layer $\mathcal{L}_q$, with $1 < p < q$.

Note that if $X \in \Delta_{\bowtie}$ and has the form

$$X = \left(0_{\mathcal{L}_1},\ldots,0_{\mathcal{L}_{p-1}},1_{\mathcal{L}_p},\ldots,1_{\mathcal{L}_r}\right),$$

then

$$W(X) = \left(0_{\mathcal{L}_1},\ldots,0_{\mathcal{L}_p},W_{p+1,p}1_{\mathcal{L}_p},\ldots,W_{r,r-1}1_{\mathcal{L}_{r-1}}\right)$$

and $W(X) \in \Delta_{\bowtie}$. Thus $W_{q,q-1}1_{\mathcal{L}_{q-1}}$ must have all coordinates zero, which is a contradiction since, by assumption, the cells in $\mathcal{L}_q$ have nonzero valency. □

Corollary 3.5 *In a FFNN, we have the following:*
(i) Two cells can synchronise only if they are in the same layer and have the same valency.
(ii) If each layer has exactly one cell then no two cells in the network can synchronize.

Proposition 3.6 *Any quotient network of a FFNN is also a FFNN.*

Proof Let G be an n-cell FFNN with layers $\mathcal{L}_1,\ldots,\mathcal{L}_r$ and $\bowtie$ be a balanced equivalence relation on the network set of cells. By Corollary 3.5 just cells in the same layer can syncronise. Thus, $\bowtie$ refines the equivalence relation with classes $\mathcal{L}_1,\ldots,\mathcal{L}_r$. Let W be the network weighted adjacency matrix and $\bowtie_i$ be restriction of $\bowtie$ to the layer $\mathcal{L}_i$, say with classes $\mathcal{I}_1,...,\mathcal{I}_{p_i}$. Then the weighted adjacency matrix of the quotient network Q has a lower-triangular block structure as in (3.4) where each nonzero submatrix $Q_{i,i-1}$ for $i \in \{2,...,r\}$ has p_i columns whose jth column is equal to the sum of the columns in $W_{i,i-1}$ associated to the cells in the class $\mathcal{I}_j$, $j \in \{1,\ldots,p_i\}$. □

Remark 3.7 It is not true that any lift of a FFNN is also a FFNN. For example, the network G of Figure 1 (left) is a lift of the network Q in Figure 1 (right) but Q is a FFNN and G is not. ◇

The next proposition is useful for the development of Algorithm 3.10 below.

Proposition 3.8 *Let G be a FFNN with layers $\mathcal{L}_1,\mathcal{L}_2,\ldots,\mathcal{L}_r$ and set $\mathcal{C} = \cup_{i=1}^r \mathcal{L}_i$. For $i = 2,\ldots,r$, denote by G_i the subnetwork of G with layers $\mathcal{L}_{i-1}$, $\mathcal{L}_i$ and containing the connections in G from the cells in the layer $\mathcal{L}_{i-1}$ to the cells in the layer $\mathcal{L}_i$. An equivalence relation $\bowtie$ on $\mathcal{C}$ refining the equivalence relation with classes $\{\mathcal{L}_1,\mathcal{L}_2,\ldots,\mathcal{L}_r\}$ is balanced for G if and only if, for $i = 2,\ldots,r$, the restriction of $\bowtie$ to $\mathcal{L}_{i-1}\cup\mathcal{L}_i$, denoted by $\bowtie|_{\mathcal{L}_{i-1}\cup\mathcal{L}_i}$, is balanced for G_i.*

Remark 3.9 Following the notation of Proposition 3.8 above, consider that the cells of G are enumerated such that the weighted adjacency matrix W of G has the lower-triangular block form as in (3.4). Observe that the submatrix $W_{i,i-1}$ corresponds to the weighted adjacency matrix of the subnetwork G_i. We establish a condition on $W_{i,i-1}$ for $\bowtie|_{\mathcal{L}_{i-1}\cup\mathcal{L}_i}$ to be balanced for the subnetwork G_i: construct $\overline{W}_{i,i-1}$ from $W_{i,i-1}$ where each column is the sum of the columns of $W_{i,i-1}$ indexed by the cells in each class of $\bowtie|_{\mathcal{L}_{i-1}}$. Then $\bowtie|_{\mathcal{L}_{i-1}\cup\mathcal{L}_i}$ is balanced for G_i if and only if two cells in $\mathcal{L}_i$ in the same $\bowtie$-class correspond to equal rows of $\overline{W}_{i,i-1}$. See for example Definition 2.1 and Proposition 2.2 of Aguiar *et al.* [2]. ◇

Based on the previous results, we describe below an algorithm that, given a FFNN, enumerates all the balanced equivalence relations for that network. By Proposition 3.8, the cells of a layer $\mathcal{L}_i$ synchronise according to a certain pattern depending on how the cells in the previous layer $\mathcal{L}_{i-1}$ are grouped into a synchrony pattern. Thus, the algorithm starts by considering the cells of the first layer and, since these cells have no input, they can synchronise according to any synchrony pattern. Then, for each possible synchrony pattern of the cells in the first layer, the algorithm determines recursively the possible synchrony patterns for the cells in $\mathcal{L}_2,\ldots,\mathcal{L}_r$.

Algorithm 3.10 Input: A FFNN with the cells enumerated such that the weighted adjacency matrix has the lower-triangular block form as in (3.4) determined by $W_{2,1},W_{3,2},\ldots,W_{r,r-1}$, where $l_i = \#\mathcal{L}_i$, for $i = 1,\ldots,r$, and each block $W_{i,j}$ is an $l_i \times l_{i-1}$ submatrix.

1. Set R to be the equivalence relation on the set of cells $\mathcal{C}$ with classes $\mathcal{L}_1$ and $\{i\}$, for $i \in \mathcal{C}\backslash\mathcal{L}_1$.
2. Let S_1 to be the set of all refinements of R.
3. Set $S_i := \emptyset, i = 2,\ldots,r$.
4. For $i = 2,\ldots,r$:
   - 4.1 For each $\bowtie$ in S_{i-1}:
     - 4.1.1 Let $k :=$ # classes $\bowtie|_{\mathcal{L}_{i-1}}$.
     - 4.1.2 Construct the $l_i \times k$ matrix $\overline{W}_{i,i-1}$ from $W_{i,i-1}$ in the following way: each column of $\overline{W}_{i,i-1}$ is the sum of the columns of $W_{i,i-1}$ indexed by the cells in each class of $\bowtie|_{\mathcal{L}_{i-1}}$.

4.1.3 Define an equivalence classe $\bowtie_i$ on the set of cells $\mathcal{L}_i$ in the following way: $p, q \in \mathcal{L}_i$, $p \bowtie_i q \Leftrightarrow$ rows p, q of $\overline{W}_{i,i-1}$ are equal.

4.1.4 For each refinement R_i of $\bowtie_i$:

4.1.4.1 Set $P_i := Join\,(\mathcal{L}_i, R_i, \bowtie)$;

4.1.4.2 $S_i := S_i \cup \{P_i\}$.

5. Output S_r.

$\diamond$

$Join\,(\mathcal{L}_i, R_i, \bowtie)$ removes the classes $\{j\}, j \in \mathcal{L}_i$ from $\bowtie$ and adds the classes of R_i; thus outputs an equivalence relation that is balanced (for the layers $\mathcal{L}_1 \cup \cdots \cup \mathcal{L}_i$), where the classes contained in $\mathcal{L}_{i+1} \cup \cdots \cup \mathcal{L}_r$ are singletons: $\{j\}, j \in \mathcal{L}_{i+1} \cup \cdots \cup \mathcal{L}_r$.

4 Synchronisation in AFFNNs

We consider now AFFNNs.

Definition 4.1 A FFNN where at least one cell in $\mathcal{L}_1$ has a self-loop is an *AFFNN*. $\diamond$

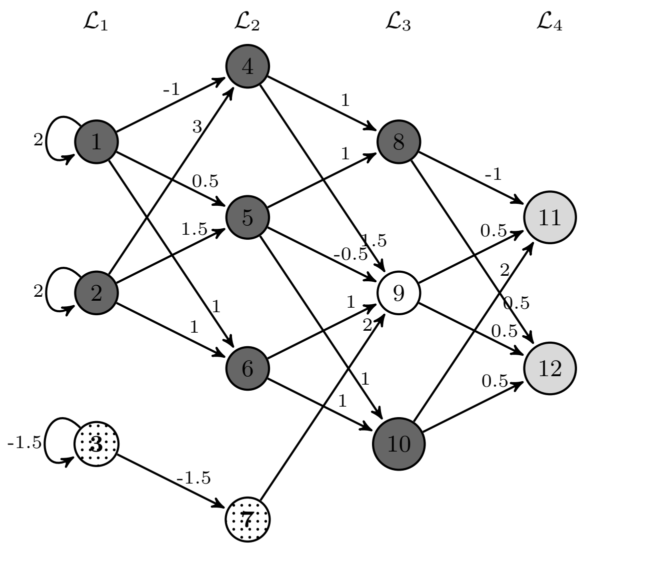

Figure 4: An AFFNN with four layers and twelve cells. The equivalence relation on the network set of cells with classes $\{1, 2, 4, 5, 6, 8, 10\}$, $\{3, 7\}$, $\{9\}$, $\{11, 12\}$ is balanced.

Example 4.2 Figure 4 shows an example of an AFFNN with 4 layers. $\diamond$

The weighted adjacency matrix of an AFFNN has a lower-triangular block structure similar to that in (3.4):

$$W = \begin{bmatrix} W_{1,1} & 0_{1,2} & \dots & 0_{1,r-1} & 0_{1,r} \\ W_{2,1} & 0_{2,2} & \dots & 0_{2,r-1} & 0_{2,r} \\ 0_{3,1} & W_{3,2} & \dots & 0_{3,r-1} & 0_{3,r} \\ \dots & \dots & \dots & \dots & \dots \\ 0_{r,1} & 0_{r,2} & \dots & W_{r,r-1} & 0_{r,r} \end{bmatrix}, \qquad (4.5)$$

where $W_{1,1}$ is a non-zero $l_1 \times l_1$ diagonal matrix describing the self-loop connections of cells in the first layer.

Two cells c, d of a network are *connected* if there is a directed path between the two cells in the network.

Definition 4.3 [20, Section 6] Let G be a network with set of cells $\mathcal{C}$ and let $c, d \in \mathcal{C}$. We call the *input subnetwork of cell* c in G, which we denote by G_c, the subnetwork of G containing all the cells in $\mathcal{C}$ that are connected to c and all the corresponding paths leading to cell c. $\diamond$

Remark 4.4 (i) Let G be a FFNN (or AFFNN) with layers $\mathcal{L}_1, \dots, \mathcal{L}_r$ and let $c_i \in \mathcal{L}_i$ for $i > 1$. Then the set of cells of the input subnetwork G_{c_i} is a subset of $\mathcal{L}_1 \cup \cdots \cup \mathcal{L}_{i-1} \cup \{c_i\}$.
(ii) An input subnetwork of a FFNN (AFFNN) is also a FFNN (AFFNN).
(iii) Let G be a FFNN (AFFNN) and $c_i \in \mathcal{L}_i$, $c_j \in \mathcal{L}_j$, with $i < j$. If c_i and c_j are connected then $\mathrm{G}_{c_i} \subset G_{c_j}$. $\diamond$

4.1 Patterns of synchrony

We characterize the patterns of synchrony for an AFFNN.

Remark 4.5 In an AFFNN, just cells in $\mathcal{L}_1$ with auto-regulation can synchronize with cells in a different layer $\mathcal{L}_i, i > 1$, since we are assuming that all cells in $\mathcal{L}_2 \cup \cdots \cup \mathcal{L}_r$ have nonzero valency. $\diamond$

Lemma 4.6 *Let G be an AFFNN with layers $\mathcal{L}_1, \dots, \mathcal{L}_r$ and consider a non spurious synchrony pattern on the network set of cells. Consider two cells $c_1 \in \mathcal{L}_1$, $c_s \in \mathcal{L}_s$, with $s > 1$, which are connected. If c_1 and c_s are synchronised then all the cells in G_{c_s} are synchronised.*

Proof Consider the weighted adjacency matrix W of G as in (4.5). Since c_1 and c_s are synchronised, then cell c_1 has a self-loop. Moreover, cells c_1 and c_s have the same valency, which is the weight of the self-loop of c_1 (Remark 2.7). We show first that all the cells in the input set of c_s (in $\mathcal{L}_{s-1}$) have to synchronize with cells c_1 and c_s. Take the vector $X \in \mathbf{R}^n$, where $x_i = 0$ if cell i is not synchronised with c_1 and c_s, and 1 otherwise. Applying W to X we have that WX has the form:

$$\begin{aligned} &WX = (Y_1, \dots, Y_r), \text{ with } Y_1 = W_{1,1}X_{\mathcal{L}_1}, \\ &\text{and } Y_i = W_{i,i-1}X_{\mathcal{L}_{i-1}}\,(i = 2, \dots, r)\,. \end{aligned} \qquad (4.6)$$

At the c_1 position of WX (taken from the vector $W_{1,1}\mathcal{L}_1$), we have the weight of the self-loop of cell c_1, which is also its valency. At the c_s position of WX (taken from the vector $W_{s,s-1}\mathcal{L}_{s-1}$) we have the sum of weights of the directed edges from cells in $\mathcal{L}_{s-1}$ to cell c_s that are also synchronised with c_1, c_s. Since c_1 and c_s are synchronised, these two entries of W at the c_1 and c_s positions must be equal and so the sum of weights in the c_s position is the valency of cell c_s (and c_1). As the synchrony pattern is not spurious, then cell c_s only receives directed edges from cells in $\mathcal{L}_{s-1}$ that are synchronised with c_s. Applying this recursively, we obtain that any directed path from a cell in $\mathcal{L}_1$ to cell c_s has to be of synchronised cells in the same synchrony class as c_1, c_s. Thus, all the cells in the subnetwork G_{c_s} are synchronised with c_1 and c_s. □

Lemma 4.7 *Let G be an AFFNN with layers $\mathcal{L}_1, \ldots, \mathcal{L}_r$ and consider a non spurious synchrony pattern on the network set of cells of G. Let $c_p \in \mathcal{L}_p$ and $c_q \in \mathcal{L}_q$, with $p < q$, be two cells that are connected. If cells c_p and c_q are synchronised then all the cells in G_{c_q} are synchronised (with c_q and c_p).*

Proof Let $c_p \in \mathcal{L}_p$ and $c_q \in \mathcal{L}_q$, with $p < q$, be two cells of G that are connected and synchronised. Consider the input subnetwork of c_p and take a cell, say c_m, such that m is the minimal integer for each $c_m \in G_{c_p} \cap \mathcal{L}_m$ and c_m is synchronized with c_p, c_q. Thus $m \leq p < q$. Fix a directed path P from the cell c_m to cell c_q through the cell c_p. Because c_m and c_q are synchronized and the synchrony pattern is not spurious, then the cell in that path belonging to $G_{c_q} \cap \mathcal{L}_{q-1}$, say c_{q-1}, must be synchronized with (at least) one cell belonging to $G_{c_m} \cap \mathcal{L}_{m-1}$, say c_{m-1}. Now, join the directed edge from c_{m-1} to c_m to P. Consider the cell in the path belonging to $G_{c_q} \cap \mathcal{L}_{q-2}$, say c_{q-2}, which has to be synchronized with some cell belonging to $G_{c_{m-1}} \cap \mathcal{L}_{m-2}$, say c_{m-2}, and join the directed edge from c_{m-2} to c_m to the path P. Continuing, we construct a directed path from a cell c_1 in $\mathcal{L}_1$ to the cell c_q passing through c_{q-m+1} where cell c_1 and c_{q-m+1} are synchronized. Thus, by Lemma 4.6 we have that all the cells in the path between c_1 and c_{q-m+1} are synchronized. But, from the choice of m, we have that $q-m+1 \geq p$, otherwise m would not be minimal as c_p belongs to the path. Thus, cell c_p is also synchronized with c_1 and c_{q-m+1}. But then we have cells in the first layer, $\mathcal{L}_1$, that are synchronised with c_q, and are connected to c_q (as there is at least the connected path P from c_p to c_q). By Lemma 4.6 we obtain that all the cells in G_{c_q} are synchronised.

□

We can now make the following conclusion:

Theorem 4.8 *Let G be an AFFNN. Consider a non spurious synchrony pattern on the set of cells of G and the associated balanced colouring. We have that the only colours that can appear sequentially repeated are the colours of the auto-regulation cells in the first layer. More concretely, given a path with first cell in $\mathcal{L}_1$ on the network, there are the following three possibilities:*
(a) all the cells have the same colour;
(b) the first cells have the same colour and all the subsequent cells have different colours;
(c) all the cells have different colours.

Proof The result follows from Lemma 4.7. Given a path, if the first cell does not synchronise with any other cell in the path then there is only the third possibility. This happens, in particular, if the first cell has no self-loop. If the first cell has a self-loop and synchronises with some cell in the path then both the first and second possibilities can occur.

□

As a consequence of Theorem 4.8 we have the following corollary that gives another necessary condition for a pattern on the cells of an AFFNN to be a pattern of synchrony.

Corollary 4.9 *Let G be an AFFNN. Consider a non spurious synchrony pattern on the set of cells of G and the associated balanced colouring. Let $c_p \in \mathcal{L}_p$ and $c_q \in \mathcal{L}_q$ be two cells that are synchronised but are not connected. Then, for each path in the input subnetwork G_{c_p} there is at least one path in the input subnetwork G_{c_q} such that the sequence of colours for the two paths is the same. Nevertheless, the number of cells with the first colour in the sequence can differ for the two paths.*

Remark 4.10 The results of this section do not hold for balanced spurious patterns. Figure 3 on the left is an example of a spurious synchrony pattern that does not lie in any of the synchrony patterns described in Theorem 4.8: note that for example cells 4 and 10 have the same colour, are connected and there are cells in the path between the two of black colour. Nevertheless, a similar result could be obtained for spurious patterns where now the colourings would include patterns as the one illustrated at the network on the left of Figure 3. The directed edges from cells 7 and 8 project in the quotient into a zero weight connection. That is, the dynamics of the cell in the $\bowtie$-class of 10 does not depend on the dynamics of the cell in the $\bowtie$-class of $7, 8$. Equivalently, this spurious pattern is balanced because it is a non spurious balanced pattern for the subnetwork of the network on the left of Figure 3 where the directed edges from cells 7 and 8 to cell 10 are ignored. ◇

The observations in the following remark are useful for the development of Algorithm 4.13 below.

Remark 4.11 Given G an AFFNN and a synchrony pattern for G associated with a non spurious balanced relation $\bowtie$ on the set of cells of G, consider the refinement $\bowtie_r$ of $\bowtie$ such that the $\bowtie_r$-classes with more than one cell are the $\bowtie$-classes with more than one cell and containing at least one cell in the first layer. Trivially, the relation $\bowtie_r$ is balanced for G and we can consider the quotient network $Q = G/\bowtie_r$. It follows from Theorem 4.8 that Q is a network where all the cells in the first layer are desynchronised. The restriction $\bowtie_q$ of $\bowtie$ to the cells of Q is a balanced relation for Q. Moreover, we have $G/\bowtie = Q/\bowtie_q$. ◇

Remark 4.12 It is not true that a quotient network of an AFFNN has to be an AFFNN. See Figure 5 for an example. ◇

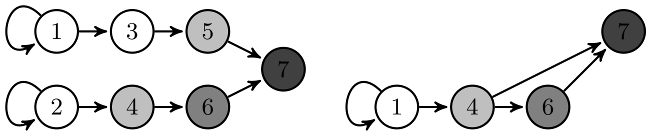

Figure 5: On the left an AFFNN with a balanced colouring. On the right the corresponding quotient which is not an AFFNN.

Based on Remark 4.11 and Theorem 4.8, we describe below an algorithm that enumerates all possible non-spurious balanced equivalence relations on the set of cells of an AFFNN. Steps $1-4$ compute the set S of all the balanced relations $\bowtie_r$ corresponding to the refinements of non-spurious balanced relations $\bowtie$ of the AFFNN such that the $\bowtie_r$-classes with more than one cell are the $\bowtie$-classes with more than one cell and containing at least one cell in the first layer. Step 5 computes the balanced relations for the quotient network associated to each relation in S and then their lift to a (non-spurious) balanced relation for the given AFFNN.

Algorithm 4.13 Input: An AFFNN with the cells enumerated such that the weighted adjacency matrix has the lower-triangular block form as in (4.5) determined by $W_{1,1}, W_{2,1}, W_{3,2}, \ldots, W_{r,r-1}$, where $l_i = \#\mathcal{L}_i$, for $i = 1, \ldots, r$, and each block $W_{i,i-1}$ is an $l_i \times l_{i-1}$ submatrix.

1. In $\mathcal{L}_1$ only the cells with the same valency can synchronize. Set $\sim_{v_1}$ to be the equivalence relation on the set of cells $\mathcal{C}$ such that cells in $\mathcal{C} \setminus \mathcal{L}_1$ are not related to any other cell and for the cells in $\mathcal{L}_1$ the relation is defined by the following: given $c, d \in \mathcal{L}_1$, we have that $c \sim_{v_1} d$ if and only if c and d have the same valency.

2. Set S_1 as the the set of all the refinements of $\sim_{v_1}$ and $i := 1$.

3. Set $i := i + 1$ and $S_i := \emptyset$:

   3.1 For each $\bowtie$ in S_{i-1}:

   3.1.1 Identify the $\bowtie$ -classes that contain at least one cell of $\mathcal{L}_{i-1}$. Set B as the set of those classes and $t := \#B$.

   3.1.1.1 Consider the subset B_1 of the $\bowtie$ -classes in B that contain at least one cell of $\mathcal{L}_1$. Set $k := \#B_1$. For each class $I_l \in B_1$, $l = 1, \ldots, k$, let v_l be the valency of the cells in I_l.

   3.1.2 Construct an $l_i \times t$ matrix $\overline{W}_{i,i-1}$ from $W_{i,i-1}$ in the following way: each column $l \in \{1, \ldots, k\}$ of $\overline{W}_{i,i-1}$ is the sum of the columns of $W_{i,i-1}$ indexed by the cells in $I_l \cap \mathcal{L}_{i-1}$.

   3.1.3 For each column $l \in \{1, \ldots, k\}$ of $\overline{W}_{i,i-1}$: identify the rows such that the element at column l is v_l and the others elements are equal to zero. Set R_l as the set of rows under these conditions.

   3.1.4 If every R_l is empty then go to step 3.1.

   3.1.5 Set $K = \{1, \ldots, k\}$.

   3.1.6 For each $l \in K$

   3.1.6.1 If $R_l \neq \emptyset$: set SP_l as the power set of R_l. For each subset in SP_l replace it by its union with I_l.

   3.1.6.2 If $R_l = \emptyset$, set $K := K \setminus \{l\}$.

   3.1.7 Set SP to be the set of all possible combinations of one subset in each SP_l, $l \in K$.

   3.1.8 For every $P \in SP$: consider the new equivalence relation $\tilde{\bowtie}$ on $\mathcal{C}$ obtained from the initial relation $\bowtie$ by removing the classes that are contained in a subset in P and adding the subsets in P as new classes. Add the new relation $\tilde{\bowtie}$ to the set S_i.

   3.2 If $i < r$ and $S_i \neq \emptyset$ then go to step 3.

4. Set $S := S_1 \cup \cdots \cup S_i$, $F_1 := S$, $L = F_1$ and $j = 1$.

5. While $F_j \neq \emptyset$:

   5.1 Set $j = j + 1$ and $F_j = \emptyset$.

   5.2 While $F_{j-1} \neq \emptyset$:

   5.2.1 Let $\bowtie$ in F_{j-1}, $F_{j-1} := F_{j-1} \setminus \{\bowtie\}$.

   5.2.2 Set $W_{\bowtie}$ to be the weighted adjacency matrix of the quotient network Q determined by $\bowtie$ and q the number of rows (columns) of $W_{\bowtie}$.

   5.2.3 Let D_Q to be the set of cells such that the off-diagonal elements of the corresponding row in $W_{\bowtie}$ are all zero.

   5.2.4 Define the equivalence relation $\sim_v$ on the set of cells of Q such that $c \sim_v d$ if and only if cells c and d are not in D_Q and the corresponding rows of the matrix $W_{\bowtie}$ are equal.

   5.2.5 For each refinement R_n of the relation $\sim_v$ excluding the trivial relation where all the classes are singletons:

   5.2.5.1 Set $P_n := Mutate(R_n, \bowtie)$;

   5.2.5.2 $F_j := F_j \cup \{P_n\}$.

   5.2.6 $L = L \cup F_j$.

6. Output L.

$\diamond$

$Mutate(R_n, \bowtie)$ removes from $\bowtie$ all the classes $\{c\}$ such that $\{c\}$ is a subset of some element of R_n and adds the classes of R_n.

5 Conclusion and future directions

One common question in neuroscience is why synchrony is so persistent in feed-forward networks. One of the strategies that has been used in that setup is to investigate how the common input stimulus (excitatory or inhibitory) of neurons in one layer affects the synchronization of the neurons of that and subsequent layers. Moreover, it is common for the propagation of synchronization along the layers to occur in a robust way. See for example [16, 22, 25].

In our approach, we associate to feed-forward neural like networks, dynamical systems that evolve with time and we ask how auto-regulation in one layer (the first layer) influences the synchrony patterns of the system. Here, the auto-regulation corresponds to the addition of auto-connections from neurons to themselves. And, robust patterns of synchrony correspond to the existence of dynamical solutions where groups of neurons (of different layers) evolve in a synchronous way for all time.

We prove that the reason for the existence of such solutions is the feed-forward network structure itself. We show how the feed-forward and auto-regulation structure force the existence of patterns that appear as a 'propagation' of the synchronization of the neurons of the first layer along the subsequent layers; taking paths in the network starting at neurons of the first layer the, synchrony pattern can in fact be characterized by the cells of the first layers (up to a certain layer, that may be the last) being synchronized and the cells of the subsequent layers (if any) being desynchronized.

One future direction is to extend this study to other classes of recurrent networks. A *recurrent network* is a FFNN with additional connections between cells of different layers forming a directed cycle. See for example the overview paper by Lipton, Berkowing and Elkan [18].

Feed-forward networks that have been studied theoretically from the bifurcation point of view are the feed-forward chains, see [12], [8], [14], [13], [24]. Another interesting direction is to explore more the study of synchrony-breaking bifurcations for general feed-forward networks.

Acknowledgments

The first two authors acknowledge partial support and the third author full support by CMUP (UID/MAT/00144/2013), which is funded by FCT (Portugal) with national (MEC) and European structural funds through the programs FEDER, under the partnership agreement PT2020.

References

[1] M. A. D. Aguiar and A. P. S. Dias. The lattice of synchrony subspaces of a coupled cell network: Characterization and computation algorithm, *Journal of Nonlinear Science*, **24** (6) (2014), 949–996.

[2] M. A. D. Aguiar, A. P. S. Dias, M. Golubitsky, M. C. A. Leite. Bifurcations from regular quotient networks: a first insight, *Physica D: Nonlinear Phenomena*, **238** (2) (2009), 137–155.

[3] P. Ashwin, G. Orosz, J. Wordsworth and S. Townley. Dynamics on networks of cluster states for globally coupled phase oscillators, *SIAM Journal on Applied Dynamical Systems* **6** (4) (2007), 728–758.

[4] F. Åström and R. Koker. A parallel neural network approach to prediction of Parkinsons Disease, *Expert systems with applications*, **38** (10) (2011), 12470–12474.

[5] R. Camera, A. W. Thomas, A. Paffi, G. d'Inzeo, F. Apollonio, F. S. Prato and M. Liberti. Effects of pulsed magnetic field on neurons: Cnp signal silences a feed-forward network model, *Neural Engineering (NER), 2013 6th International IEEE/EMBS Conference on*, (2013), 223–226.

[6] T. Dash, T. Nayak and R. R. Swain. Controlling Wall Following Robot Navigation Based on Gravitational Search and Feed Forward Neural Network, *Proceedings of the 2nd International Conference on Perception and Machine Intelligence*, (2015), 196–200.

[7] M. Diesmann, M.-O. Gewaltig and A. Aertsen. Stable propagation of synchronous spiking in cortical neural networks, *Nature*, **402** (1999), 529–533.

[8] T. Elmhirst and M. Golubitsky. Nilpotent Hopf bifurcations in coupled cell systems, *SIAM Journal on Applied Dynamical Systems*, **5** (2) (2006), 205–251.

[9] M. Field. Combinatorial dynamics, *Dynamical Systems*, **19** (3) (2004), 217–243.

[10] M. Field. Heteroclinic networks in homogeneous and heterogeneous identical cell systems, *Journal of Nonlinear Science*, **25** (3) (2015), 779–813.

[11] S. Goedeke and M. Diesmann. The mechanism of synchronisation in feed-forward neural networks, *New Journal of Physics*, **10** (2008), 015007.

[12] M. Golubitsky, M. Nicol and I. Stewart. Some curious phenomena in coupled cell networks, *Journal of Nonlinear Science*, **14** (2) (2004), 207–236.

[13] M. Golubitsky and C. Postlethwaite. Feed-forward networks, center manifolds, and forcing, *Discrete and Continuous Dynamical System*, **32** (8) (2012), 2913–2935.

[14] M. Golubitsky, L. Shiau, C. Postlethwaite and Y. Zhang. The feed-forward chain as a filter-amplifier motif, *Coherent behavior in neuronal networks*, (2009), 95–120.

[15] M. Golubitsky, I. Stewart, and A. Török. Patterns of synchrony in coupled cell networks with multiple arrows, *SIAM Journal on Applied Dynamical Systems*, **4** (1) (2005), 78–100.

[16] W. Jermakowicz, X. Chen, I. Khaytin, C. Madison, Z. Zhou and M. Bernard, A.B. Bonds,V. Casagrande. Is Synchrony a reasonable coding strategy for visual areas beyond V1 in primates?, *Journal of Vision*, **7** (9) (2007), 325-325a.

[17] S. Jahnke, R.-M. Memmesheimer and M. Timme. Propagating synchrony in feed-forward networks, *Frontiers in Computational Neuroscience*, **7** (2013), Article 153.

[18] Z.C. Lipton, J. Berkowitz and C. Elkan. A critical review of recurrent neural networks for sequence learning, *arXiv preprint arXiv:1506.00019*, (2015).

[19] F. S. Neves and M. Timme. Computation by switching in complex networks, *Physical Review Letters* **109** (2012), 01870.

[20] E. Nijholt, B. Rink and J. Sanders. Graph fibrations and symmetries of network dynamics, http://arxiv.org/abs/1410.6012 (2014).

[21] T. Nowotny and R. Huerta. Explaining synchrony in feed-forward networks: Are McCulloch-Pitts neurons good enough?, *Biological Cybernetics*, **89** (4)(2003), 237–241.

[22] A. D. Reyes. Synchrony-dependent propagation of firing rate in iteratively constructed networks in vitro, *Nature Neuroscience* **6** (2003), 593–599.

[23] A. D. Reyes. Experimental and Theoretical Analyses of Synchrony in Feed-forward Networks, *Computational Neuroscience in Epilepsy* (2008), 304–316.

[24] B. W. Rink and J.A. Sanders. Amplified Hopf bifurcations in feed-forward networks, *SIAM Journal on Applied Dynamical Systems*, **12** (2) (2013), 1135–1157.

[25] I. Segev. Synchrony is stubborn in feedforward cortical networks, *Nature Neuroscience* **6** (2003), 543–544.

[26] E. Şenyiğit, M. Düğenci, M. E. Aydin and M. Zeydan. Heuristic-based neural networks for stochastic dynamic lot sizing problem, *Applied Soft Computing*, **13** (3) (2013), 1332–1339.

[27] I. Stewart, M. Golubitsky and M. Pivato. Symmetry groupoids and patterns of synchrony in coupled cell networks, *SIAM Journal on Applied Dynamical Systems*, **2** (4) (2003), 609–646.

[28] J. Szkoła, K. Pancerz and J. Warchoł. Recurrent neural networks in computer-based clinical decision support for laryngopathies: an experimental study, *Computational intelligence and neuroscience*, **2011** (2011), 1–8.

[29] M. Tsodyks, A. Uziel and H. Markram. Synchrony Generation in Recurrent Networks with Frequency-Dependent Synapses, *The Journal of Neuroscience*, **20** (2000), RC50.