Results: Building the Comparation network and generation of equivalence relation.

Qualitative Modelling (Comparation Network showing a relationship).

One view of building a neural network performing the process of **comparation** is to find a network exhibiting the relation properties: reflexive, symmetric and transitive. This is the view we take. This is because at the stage of the process of comparation in the **Verstandes Actus** it performs logical comparison with obscure (not conscious) parástase. Comparison made by identifying particular equivalence classes implies the process is working with conscious parástase. Thus the first postulate is,

Proposition 1: Comparation in the synthesis of equivalence relation exhibits the reflexive and symmetric relations.

Such a network is said to synthesize a compatibility relation. As was discussed earlier, comparison always involves a minimum of two items or objects. Thus,

Proposition 2: The comparation process in Verstandes Actus involves a minimum of two inputs.

Thus, the basic model will involve logical comparison of two inputs. For convenience of illustration let each input appear as a retinal-map of pixels as shown below.

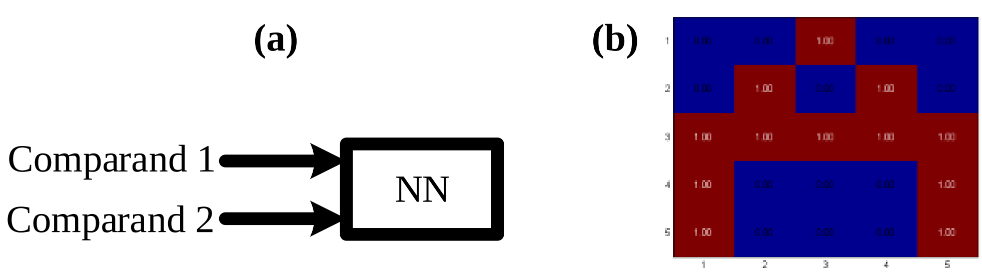

Figure 5.1. The basic model (a) receiving two comparands, each represented as retinal-map matrix sized 5 x 5.

The comparation process is in facet-B and hence is in the mathematical domain. The neural network behavior determines the relationship but does so without any a priori objective knowledge. Hence, the determination (relationship or not) is made by feature definer and detector which is automatically generated. We do not build them into the model. Hence the third postulate is,

Proposition 3: Comparation compares features that are dynamically generated axiomatic features.

It was mentioned earlier that the comparation process works with obscure parástase (input) and its determination (output) is also an obscure parástase. **TRJ** (teleological reflective judgment) judges **expedience**. This gives us,

Proposition 4: The process of comparation is judged expedient when the act is not contrary to the **categorical imperative**, which regulated to achieve a state of equilibrium.

During the process of comparison when the neural network model reaches a reverberating or resonant state, we will say the model is in steady-state condition. We will consider a neural network in such a state to be in equilibrium and hence may be judged expedient.

In embedding field theory (EFT) there is a family of known and also yet to be discovered neural networks which can reach a resonant state [Wells, 2010]. They are called adaptive resonance theory (ART) networks. A typical anatomy of an ART resonator (ART-R) is shown in figure 5.2a. Based upon this basic anatomy the network model has two resonators, such that each receives a respective input element. Hence, the model receives two comparands (Fig.5.2b). However, for the process of comparation the two basic anatomies must interact and hence cannot be isolated subsystems (Fig.5.2c). Thus,

Proposition 5: The sub-processes for respective comparands are united within the process of comparation.

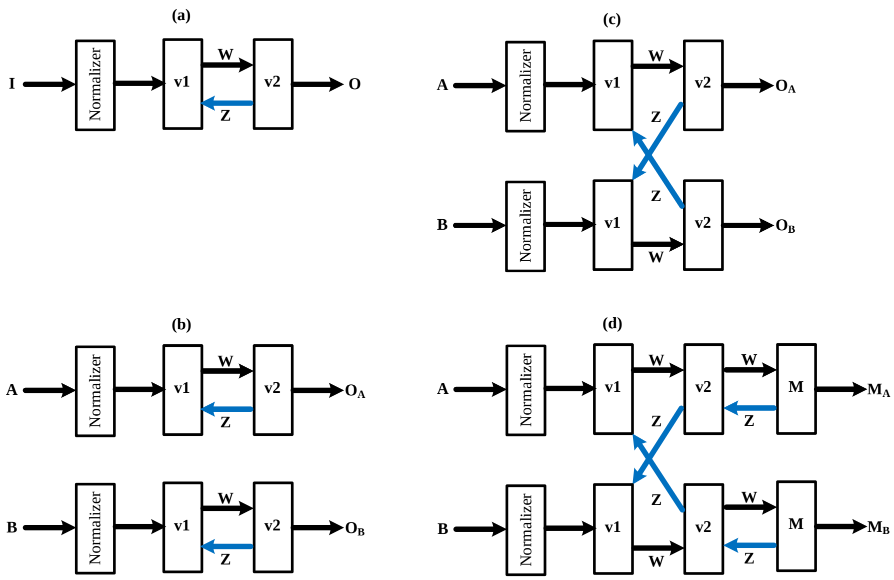

Figure 5.2. Illustration of the evolution from the basic ART-R (a) to the minimal neural network (d).

The basic ART-R (a) has a normalizer layer which then sends normalized inputs to the v1 layer. The v1 and v2 layers interact such that the nodes in v2 layers receives instar weights (W) and send out outstar weights to v1 layer. The interaction therefore plays a major role in achieving resonance.

Each ART just receiving respective comparand input (b) is insufficient. (c) Shows one approach to interaction between the two sub-networks. Here, the outstar-weighted output of the v2 layer of a sub-network feeds into the v1 layer of the other sub-network.

The comparation network is judged for expediency by interacting with M layer. Thus, this addition in the network proxies for part of TRJ. This also means that **nous-soma** connection is made. It should be noted that M_A and M_B are not motor responses but thought of as pre-motor images.

An obvious question for the above model is "how do we know that the process of comparation is expedient?" The model must have this property for at least two reasons.

Firstly, since expediency is a **judicial imperative** the determination (relationship) by comparation will have no *practical* purpose if it is not expedient. Secondly, this is judged by TRJ which is a mediator between **nous** and **psyche** co-organizations. This means that the model must have link to **motoregulatory expression**. Thus,

Proposition 6: The process of comparation is judged by TRJ (teleological reflective judgment).

In the model (Fig.5.2c) the judgment for resonance and hence expediency is evaluated by the addition to two more fields (Fig.5.2d). These additions are proxies for motoregulatory *emotivity* such that when they reach steady-state the process of comparation may be judged expedient. It should be noted that there is no longer a clear distinction between the process of comparation and the ability to judge expediency (a functional part of TRJ).

Comparation is a process within **sensibility** but sensibility does not judge, does not confuse and does not deceive [Kant, 1798]. Recall that the **synthesis in sensibility** is transcendental, i.e., necessary for the possibility of experience. Thus the activities of the motor end of the network do not represent motor response of the **OB** but are akin to pre-motor activities. We may therefore consider that if the steady-state values of the pre-motor responses are within a solution set, the comparands have a relationship. Thus,

Proposition 7: The relationship determined by comparation is *practical* in the sense that it is not contrary to the OB’s categorical imperative.

Thus meaning of the determined relationship is purely *practical*.

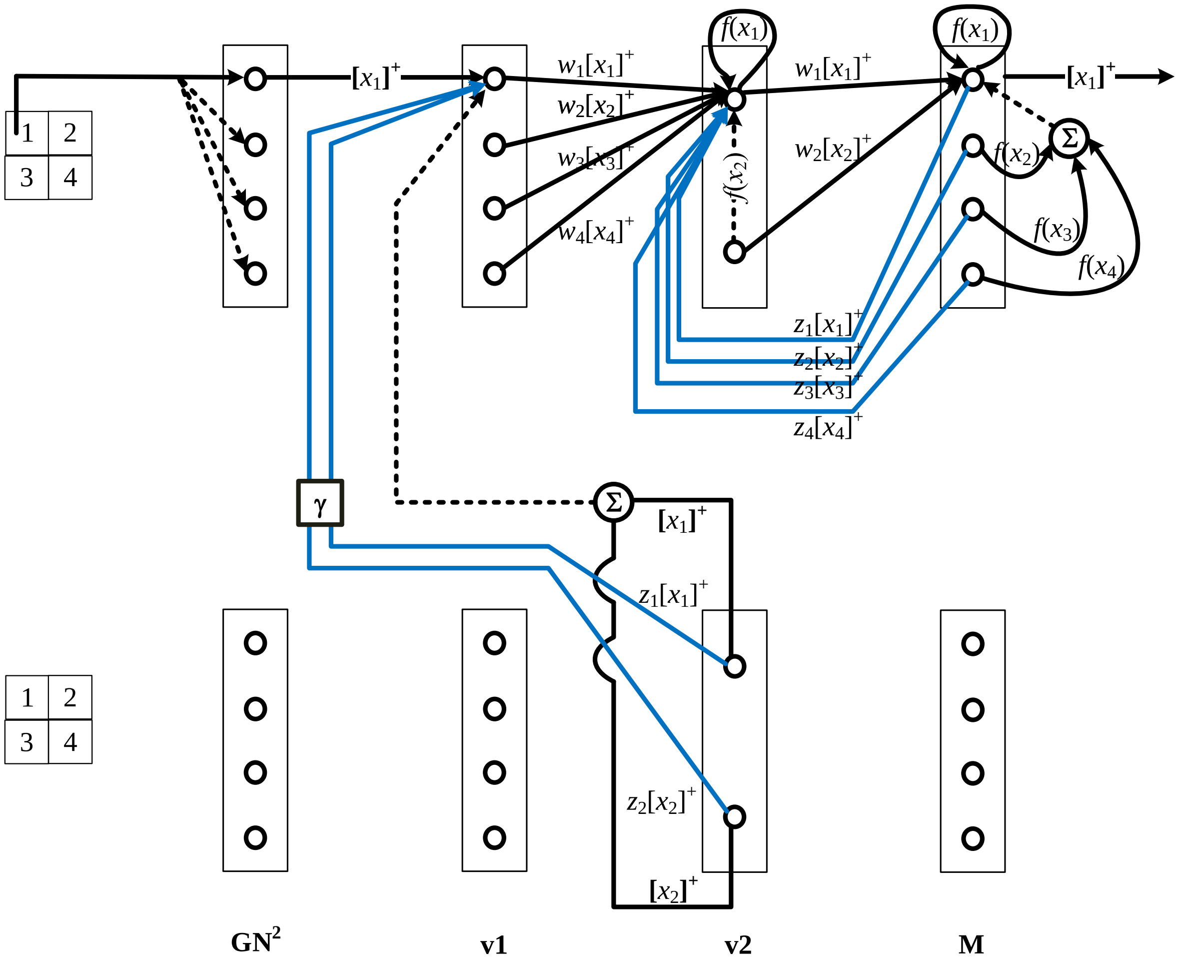

Figure 5.3. A more detailed view of the minimal neural network (Fig.5.2d). The illustration shows the connection of a single node in each layer. Given a retinal-map matrix sized 2 X 2, the first pixel excites (solid line) the first GN^2 node but inhibits (dashed lines) other nodes (of respective sub-network). The first v1 node (SNI^4) then receives excitatory inputs from the first normalized input and outstar-weighted output of all v2 nodes (other sub-network) with gain γ (solid blue line). The inhibition (dashed line) is the sum of these un-weighted v2node outputs.

The first v2 node (SNI^3) receives excitatory inputs from instar-weighted output of all v1 nodes (same sub-network), outstar-weighted output of all M nodes within the same sub-network (solid blue line) and from itself passed through the activation function ($f(\cdot)$). The rest of the v2 nodes within the same sub-network passed through $f(\cdot)$ are added to form the inhibitory input to the first v2 node. 0-1 distribution is enforced for v2 layer.

The first M node is the same kind of SNI^3 as the v2 nodes but without 0-1 distribution. Thus, its excitatory and inhibitory inputs follows a similar pattern as above but with different connectivity. The node does have any excitatory input from outstar-weighted output.

Note that *x*'s and weights *w*'s or *z*'s are labeled the same for each layer for simplicity. However, they are quantitatively different and hence don't represent for instance the same excitation.

Quantitative Model (Comparation)

The quantified (detailed) view of the figure 5.2d is shown in figure 5.3. The ART network has nodes which in general are called shunting node instars (SNI). There are numerous types of SNI's, each having a different behavior from another [Wells, 2010]. This is consistent with EFT and MMA. The general description is given above (Fig. 5.3). Below are the mathematical expressions specific to the model.

The inputs or comparands are first normalized by normalizers called a Grossberg Normalizer-2 (GN^2) [Wells, 2010]. The practical importance of having a normalizer in ART networks is elaborated in Wells' text [Wells, 2010]. The i^{th} GN^2 node is given by the differential equation,

$$\dot{x}_i^{\mathrm{GN}} = -A_o x_i^{\mathrm{GN}} + (B_o - x_i)I_i - (C_o + x_i^{\mathrm{GN}}) \sum_{\forall k \neq i} I_k,$$

where, A_o, B_o and $C_o$are non-zero parameters such that, $B_o = (n-1)C_o$. As mentioned above the comparand is a retinal-map and hence they are matrix of I_i pixels. There is a corresponding GN^2 node for each pixel. This means there will be a total of n GN^2 nodes for a retina matrix of a rows and b columns (n = $a \cdot b$). In the above equation, $I = \sum_{\forall k} I_k$ and $\omega_i = \frac{I_i}{I}$.

The normalized inputs are then received by respective v1 layer. Like the normalizer layer the v1 layer has n nodes. This layer is comprised of SNI's called SNI^4 [Wells, 2010]. It's i^{th} node is given by,

$$\dot{x}_i^{\mathrm{v1}} = -A_1 x_i^{\mathrm{v1}} + (B_1 - x_i^{\mathrm{v1}})J_i^+ - (D_1 + x_i^{\mathrm{v1}})J^-,$$

where, A_1, B_1 and D_1 are non-zero parameters. The excitatory J_i^+ and inhibitory J^- inputs are given by,

$$J_i^+ = x_i^{GN} + \gamma Z_{\text{v2 to v1}}[\vec{\mathbf{x}}^{\text{v2}}]^+ \quad \text{and} \quad J^- = \sum_{\forall k}[x_k^{\text{v2}}]^+,$$

where, γ is a non-zero parameter. The Heaviside extractor $[\,\cdot\,]^+$ is given by,

$$[H]^+ = \begin{cases} H \; if \; H \geq 0 \\ 0 \; else \end{cases}.$$

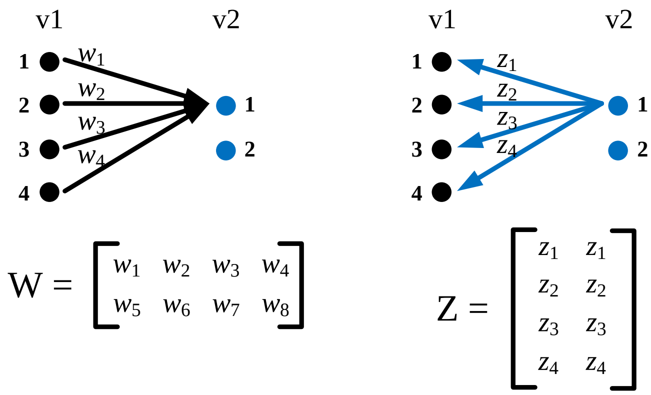

Figure 5.4. Illustration of instar (W) and oustar (Z) weights. Notice that the first row elements of W matrix corresponds to instar weights going into first v2 node (blue). But first column elements of Z matrix corresponds to outstar weights going out of the first v2 node.

$Z_{\text{v2 to v1}}$ is the outstar weight matrix from v2 layer to v1. They go hand in hand with the instar weight matrix from v1 to v2, $W_{\text{v1 to v2}}$ (Fig.5.4). They are given by the form,

$$W_{\text{v1 to v2}}(t+1) = W_{\text{v1 to v2}}(t) - \eta \cdot \left[(W_{\text{v1 to v2}}(t) - [\vec{\mathbf{x}}^{\text{v1}^{\text{T}}}]^+) \cdot [\vec{\mathbf{x}}^{\text{v2}}]^+\right] \text{ and}$$
$$Z_{\text{v2 to v1}}(t+1) = Z_{\text{v2 to v1}}(t) - \eta \cdot [(Z_{\text{v2 to v1}}(t) - [\vec{\mathbf{x}}^{\text{v1}}]^+) \cdot [\vec{\mathbf{x}}^{\text{v2}}]^+],$$

where, η is the adaptation parameter. The property of outstar weights in an ART network is that, the weights from the winning v2 nodes is an exemplar of the input.

The nodes in the v1 layer interact with v2 layer nodes. But unlike the v1 layer, the number of nodes in v2 is not restricted by the size of the retina matrix. This also means that the number of nodes in the succeeding M layer is unrestricted. SNI[3] was chosen for the v2 nodes. They are expressed by,

$$\dot{x}_i^{\mathrm{v2}} = -A_2 + (B_2 - x_i^{\mathrm{v2}})\xi_i^{Ex} - (x_i^{\mathrm{v2}} + D_2)\xi_i^{In},$$

where, A_2, B_2 and D_2 are non-zero parameters. The excitatory ξ_i^{Ex} and inhibitory ξ_i^{In} inputs are given by,

$$\xi_i^{Ex} = W_{\mathrm{v1\ to\ v2}} \cdot [\vec{\mathbf{x}}^{\mathrm{v1}}]^+ + Z_{\mathrm{M\ to\ v2}} \cdot [\vec{\mathbf{x}}^{\mathrm{M}}]^+ + f(x_i^{\mathrm{v2}}) \text{ and}$$
$$\xi_i^{In} = \sum_{\forall k \neq i} f(x_k).$$

It has been observed that adaptation in most ART networks necessitates a 0-1 distribution in the v2 layer [Wells, 2010]. Therefore, a 0-1 distribution is enforced for the vector, $\vec{\mathbf{x}}^{\mathrm{v2}}$ and the winning v2 node controls network adaptation.

The function $f(\cdot)$ is called the activation shape function. It is given by,

$$f(y) = \begin{cases} 0 & if\ y < 0 \\ \lambda \cdot y & if\ y \leq u^{(1)} \\ g_{max} \cdot y & if\ u^{(1)} < y \leq u^{(2)} \\ g_{max} \cdot u^{(2)} & if\ y > u^{(2)} \end{cases}.$$

where $u^{(1)}$ and $u^{(2)}$ are parameters for the activation shape function. The contrast enhancement capability of some ART networks may attenuate the absolute levels of x_i signals. Proper scaling will not alter the performance of the network apart from limiting the attenuation of the absolute levels [Wells, 2010]. Using the B parameter as the reference this is done here as,

$$g_{max} = B \cdot g_{max}^{unscaled},$$

$$u^1 = \frac{\left(B - \frac{A}{g_{max}}\right)}{\left(1 - \frac{A}{g_{max}}\right)} \cdot u_{unscaled}^1,$$

$u^2 = B \cdot u^2_{unscaled}$ and

$\lambda = \frac{g_{max}}{u^1}$.

where, $g^{unscaled}_{max}$, $u^1_{unscaled}$ and $u^2_{unscaled}$ are the unscaled parameters.

The nodes for the M layer use SNI[3] and hence is similar to the v2 layer. However its parameters may be different and because the connections are different the expression of excitatory and inhibitory input are different.

Therefore, the minimal neural network is given by,

$$x_i^{\text{M6}}(t+h) = \left[1 - h \cdot \left(A_{\text{M}} + \xi_i^{Ex\_\text{M6}} + \xi_i^{In\_\text{M6}}\right)\right] \cdot x_i^{\text{M6}} + h \cdot \left(B_{\text{M}} \cdot \xi_i^{Ex\_\text{M6}} - D_{\text{M}} \cdot \xi_i^{In\_\text{M6}}\right),$$
$$x_i^{\text{M5}}(t+h) = \left[1 - h \cdot \left(A_{\text{M}} + \xi_i^{Ex\_\text{M5}} + \xi_i^{In\_\text{M5}}\right)\right] \cdot x_i^{\text{M5}} + h \cdot \left(B_{\text{M}} \cdot \xi_i^{Ex\_\text{M5}} - D_{\text{M}} \cdot \xi_i^{In\_\text{M5}}\right),$$
$$x_i^{\text{v2\_4}}(t+h) = \left[1 - h\left(A_2 + \xi_i^{Ex\_4} + \xi_i^{In\_4}\right)\right]x_i^{\text{v2\_4}} + h\left(B_2\xi_i^{Ex\_4} - D_2\xi_i^{In\_4}\right),$$
$$x_i^{\text{v2\_3}}(t+h) = \left[1 - h\left(A_2 + \xi_i^{Ex\_3} + \xi_i^{In\_3}\right)\right]x_i^{\text{v2\_3}} + h\left(B_2\xi_i^{Ex\_3} - D_2\xi_i^{In\_3}\right),$$
$$x_i^{\text{v1\_2}} = \frac{B_1 J_i^+ - D_1 J^-}{A_1 + J_i^+ + J^-} \quad \text{and}$$
$$x_i^{\text{v1\_1}} = \frac{B_1 J_i^+ - D_1 J^-}{A_1 + J_i^+ + J^-}.$$

Rather than numerically solving difference equations, steady-state solution is used for the v1 layer. This is because the v1 layer in ART is considered to be faster than v2 layer. The weights (instar & outstar) for interaction between v1 – v2 and v2 – M are adaptive. The respective expressions for the excitation and inhibitory inputs are,

$$\xi_i^{Ex\_\text{M6}} = W_{\text{v2\_4 to M6}} \cdot [\vec{\mathbf{x}}^{\text{v2\_4}}]^+ + f(x_i^{\text{M6}}), \quad \xi_i^{In\_\text{M6}} = \sum_{\forall k \neq i} f(x_k^{\text{M6}});$$
$$\xi_i^{Ex\_\text{M5}} = W_{\text{v2\_3 to M5}} \cdot [\vec{\mathbf{x}}^{\text{v2\_3}}]^+ + f(x_i^{\text{M5}}), \quad \xi_i^{In\_\text{M5}} = \sum_{\forall k \neq i} f(x_k^{\text{M5}});$$
$$\xi_i^{Ex\_4} = W_{\text{v1\_2 to v2\_4}} \cdot [\vec{\mathbf{x}}^{\text{v1\_2}}]^+ + Z_{\text{M6 to v2\_4}} \cdot [\vec{\mathbf{x}}^{\text{M6}}]^+ + f(x_i^{\text{v2\_4}}), \quad \xi_i^{In\_4} = \sum_{\forall k \neq i} f(x_k^{\text{v2\_4}});$$
$$\xi_i^{Ex\_3} = W_{\text{v1\_1 to v2\_3}} \cdot [\vec{\mathbf{x}}^{\text{v1\_3}}]^+ + Z_{\text{M5 to v2\_3}} \cdot [\vec{\mathbf{x}}^{\text{M5}}]^+ + f(x_i^{\text{v2\_3}}), \quad \xi_i^{In\_3} = \sum_{\forall k \neq i} f(x_k^{\text{v2\_3}});$$

$J_i^{\text{v1\_2+}} = x_i^{\text{GN\_P2}} + \gamma Z_{\text{v2\_3 to v1\_2}}[\vec{\mathbf{x}}^{\text{v2\_3}}]^+, \quad J^{\text{v1\_2-}} = \sum_{\forall k}\left[x_k^{\text{v2\_3}}\right]^+$ and

$J_i^{\text{v1\_1+}} = x_i^{\text{GN\_P1}} + \gamma Z_{\text{v2\_4 to v1\_1}}[\vec{\mathbf{x}}^{\text{v2\_4}}]^+, \quad J^{\text{v1\_1-}} = \sum_{\forall k}\left[x_k^{\text{v2\_4}}\right]^+.$

Finally the normalization is done only once for a given pattern. Thus its steady-state form is,

$x_i^{\text{GN\_P1}} = \frac{nC_oI}{A_o+I}\left(\omega_i - \frac{1}{n}\right)$ and $x_i^{\text{GN\_P2}} = \frac{nC_oI}{A_o+I}\left(\omega_i - \frac{1}{n}\right).$

The synthesis of sensibility is the noetic process that synthesizes **apprehension** in **consciousness**. Thus the OB has no conscious experience of these acts of synthesis. In other words, comparation does not have memory. Weight changes in W and Z are elastic modulations. This implies,

Proposition 8: The process of comparation does not remember its past actions.

Hence, the states of the instar and outstar weights of the above minimal neural network are not stored for succeeding acts of logical comparison. Therefore this network does not have the problem of stability-plasticity trade-off encountered in some ART networks [Wells, 2010].

Behavior of the proposed minimal neural network.

A set of uppercase English letters was used for observing the behavior of the network (Fig.5.5). The parameters used in the simulation are shown in figure 5.6. The set-membership paradigm was employed to consider whether the network determines a relationship between alphabets (or patterns). Thus identification of feasible set solutions rather than a single “point” solution is considered.

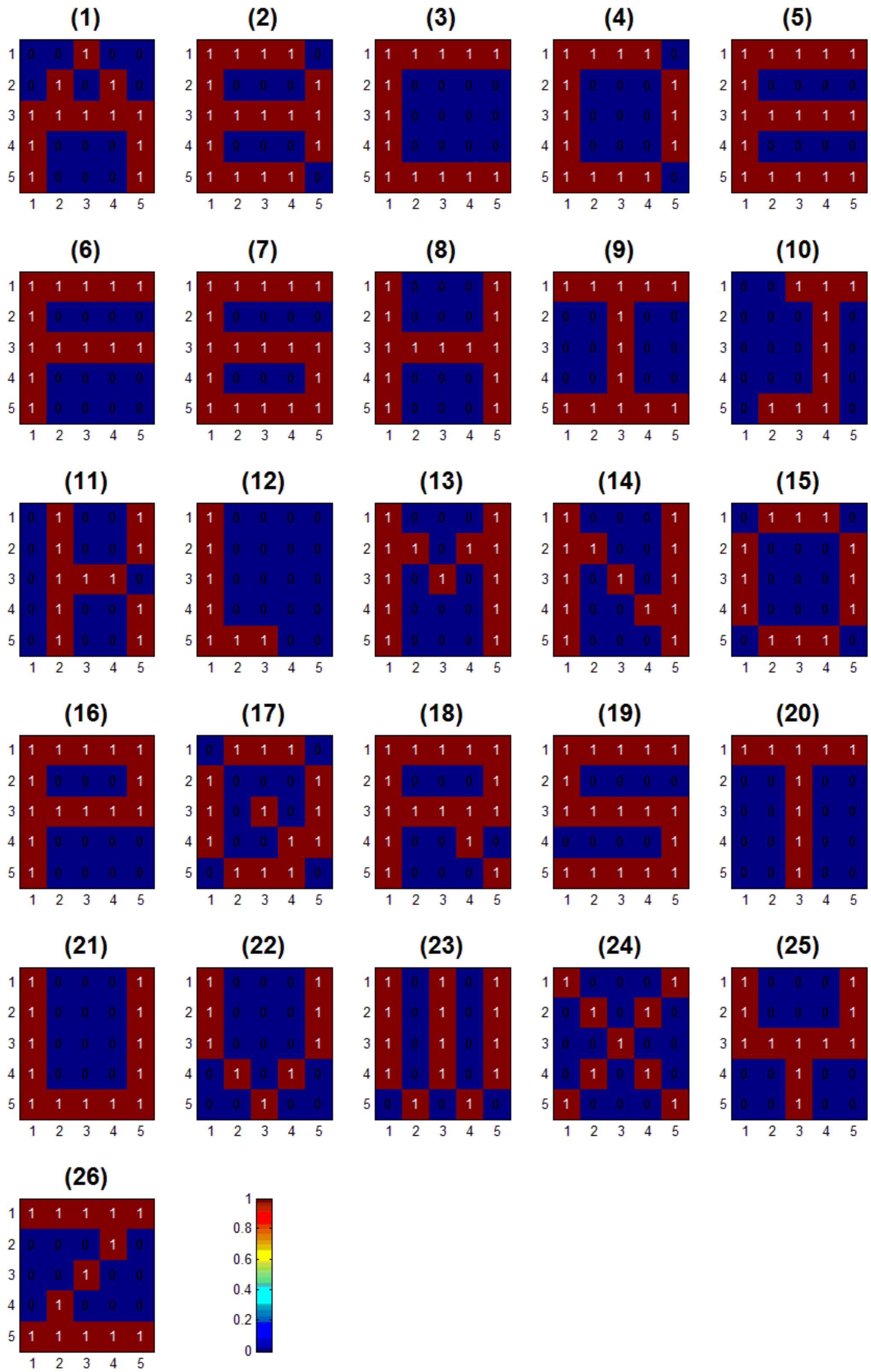

Figure 5.5. Set of upper-case English letters used for observing the network behaviors. Each alphabet is represented by the respective placement of 1's in a background of 0's.

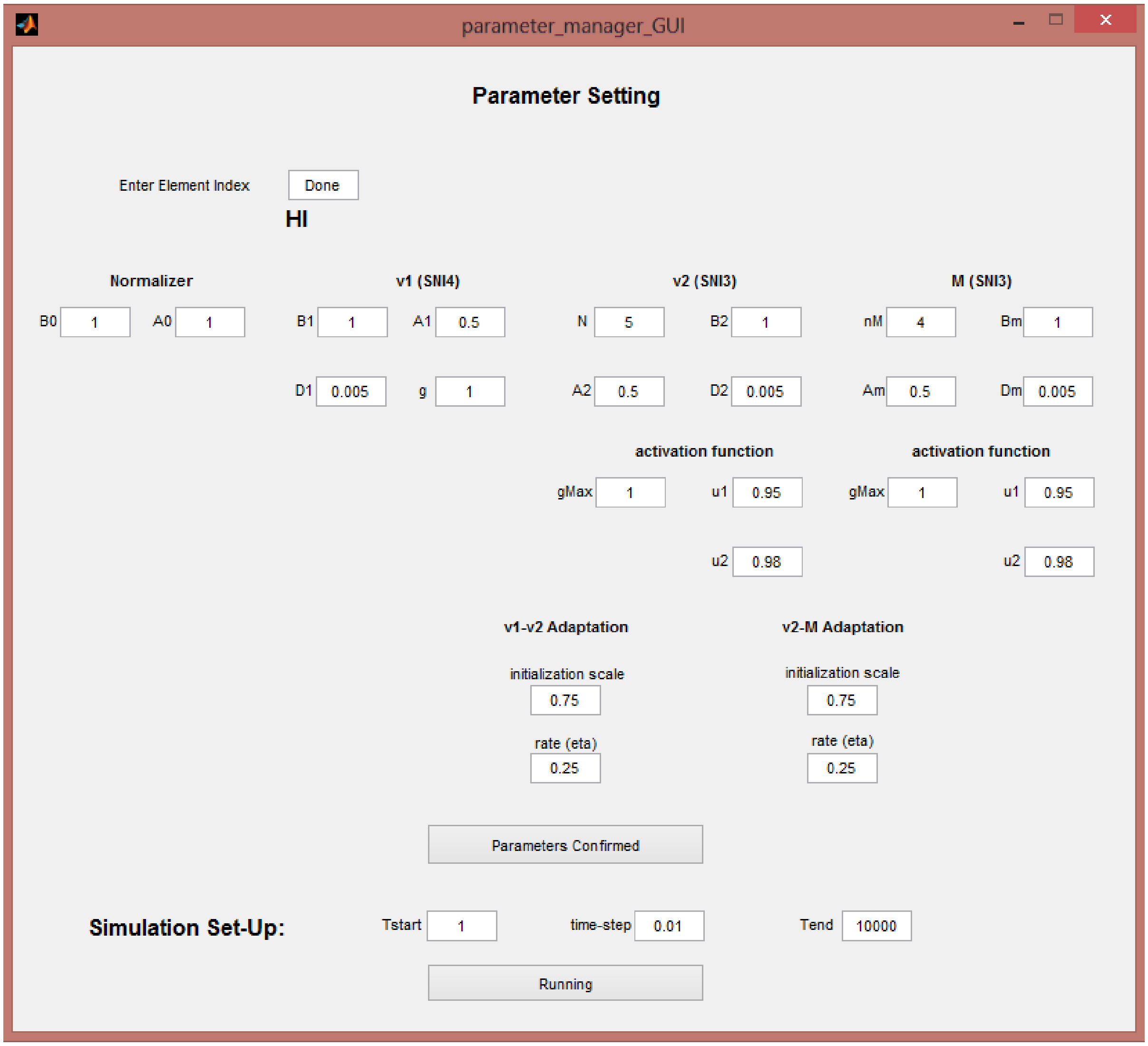

Figure 5.6. Parameters for the simulation. For a particular pattern-combo, the two desired patterns are chosen by entering their indices ranging from [1, 26] (Fig.5.5). The respective parameters are then entered. Note that g in v1 stands for gamma (Fig.5.3) and the activation function parameters are the unscaled values. The scaling is done as described above within the main code. Both the sub-networks have same parameter values.

Though the weights of the network are not stored they are initialized for each run of pattern-combo. This is done by setting the outstar weight matrix elements at zero but the elements of the instar weight matrices are set at some very small (non-zero) random real number. The latter amount is given by initialization scale.

As governed by the functional principles of ART networks [Wells, 2010], weight adaptation occurs after resonance of the network.

Finally, the time-step for the difference equations was chosen to be 0.01 and simulation was done for 10,000 iterations.

Network testing involved pattern-combos (two patterns in a combo such that each pattern is picked from the set of letters (Fig.5.5)). It was found that the network determined relationships between some patterns and not for others. We say that the network determines a relationship between the patterns if the steady-state value (final iteration) of the M layer outputs fall within the same solution set.

Figures 5.7 and 5.8 show that relationship was found between patterns H and I. The first figure, with same graduation on the ordinate, demonstrates the relative difference in the activity of a particular node of respective layer. The shape of each activity is accentuated in the second figure, which no longer has the same graduations along the ordinate (unscaled). However determination of relationship depends on the expediency judged by TRJ and so the region of interest is the steady-state levels of the M-layer outputs. The output level in both figures indicate that they fall within the solution-set. This is confirmed further in the table inset of figure 5.8 showing the magnitude of the steady-state node activity for all the four nodes in respective M-layer.

Figure 5.9 shows that the network has the ability of finding no relationship between patterns (B and R in this particular case). Notice that the plots have different graduations along the ordinate. The inset table confirms that the steady-state M-layer outputs are not within the solution-set. It shows that the steady-state M-layer output for the right sub-network (receiving pattern R) is about 2 x that of the left sub-network (receiving pattern B).

For the sake of consistency the time-series figure will show plots with different graduation scales along the ordinate. The determination of whether the M-layer outputs fall within a solution-set may be confirmed from the inset table.

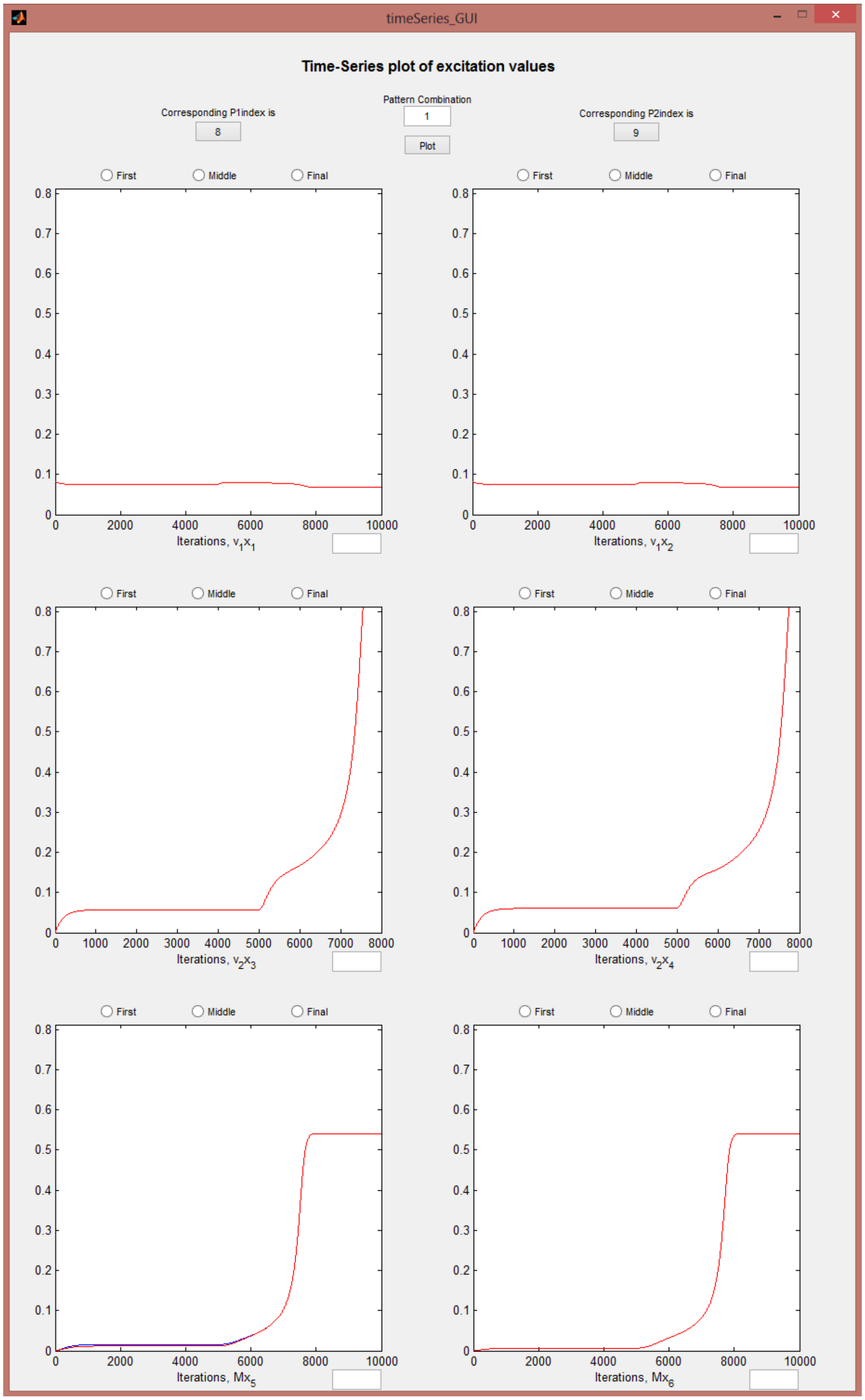

Figure 5.7. Time-series (scaled) plot for patterns H (left sub-network) and I (right sub-network). Scaled implies that the plots have same graduation in the ordinate. This plot shows the response magnitudes between layers in the top (left column) an bottom (right column) sub-networks.

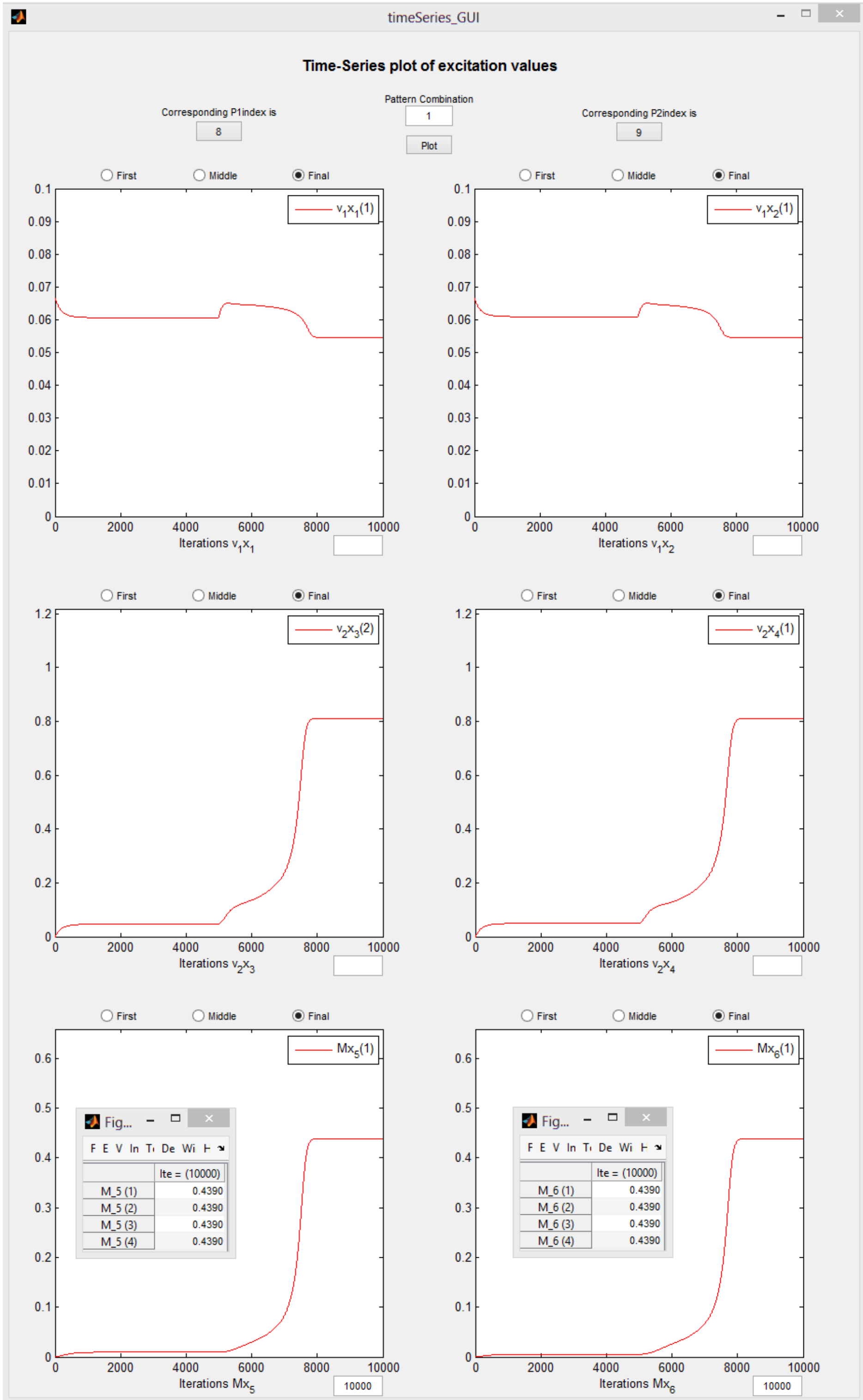

Figure 5.8. Time-series (unscaled) plot of Fig.5.7. Notice that the graduation in the ordinate are no longer the same. This shows the plot of 1st v1 nodes, 2nd v2 (winner, left sub-network), 1st v2 (winner, right sub-network) and 1st M nodes. The table in the bottom sub-plots shows the steady-state (at 10,000 iteration by entering it in the edit box) of the 4 nodes in respective M layer.

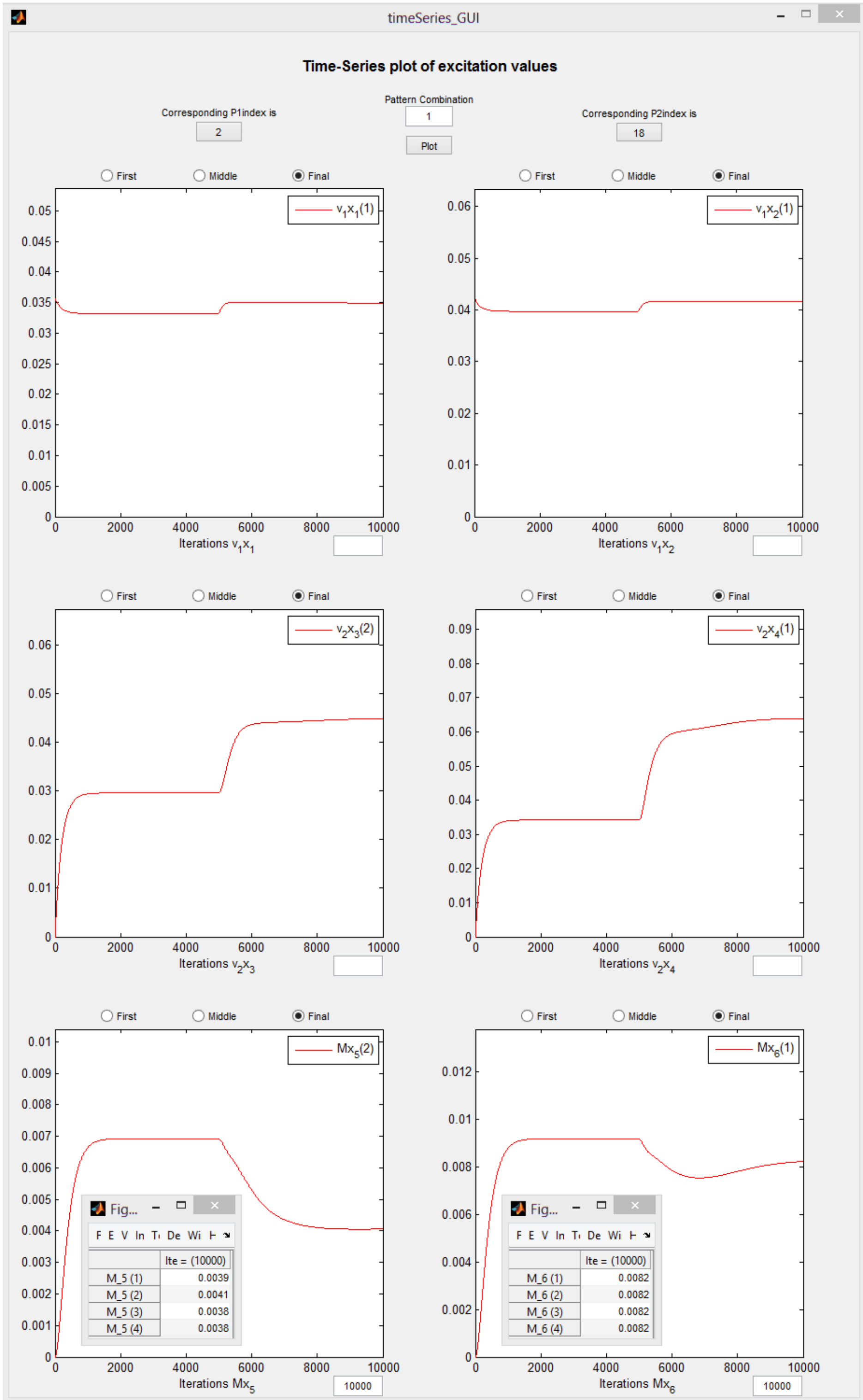

Figure 5.9. Time-series (unscaled) plot for patterns B (left sub-network) and R (right sub-network). This shows the plot of 1st v1 nodes, 2nd v2 (winner, left sub-network), 1st v2 (winner, right sub-network) and 2nd & 1st M nodes. The table in the bottom sub-plots shows the steady-state (at 10,000 iteration by entering it in the edit box) of the 4 nodes in respective M layer.

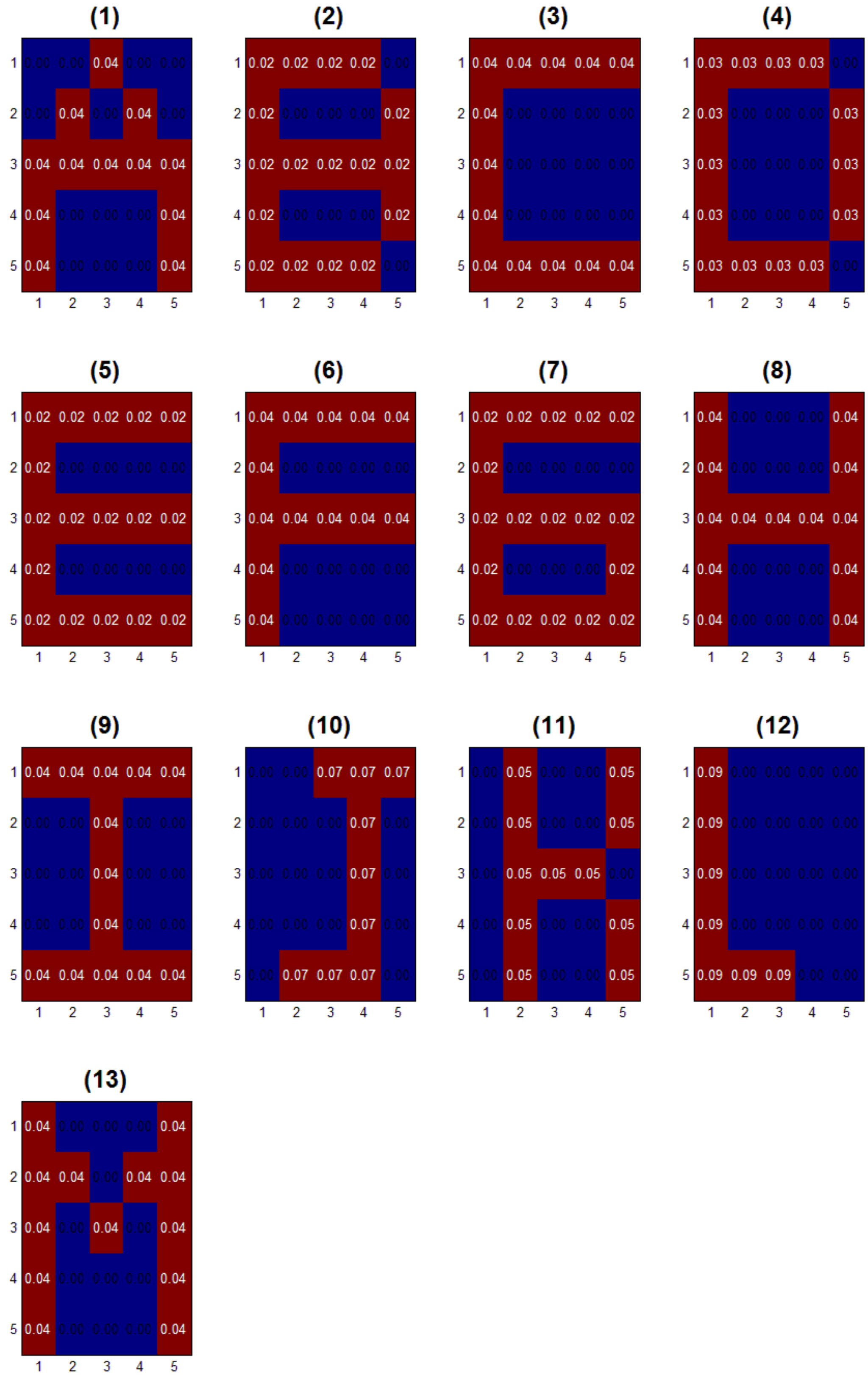

Figure 5.10. Normalized patterns (A to M) of Fig.5.5.

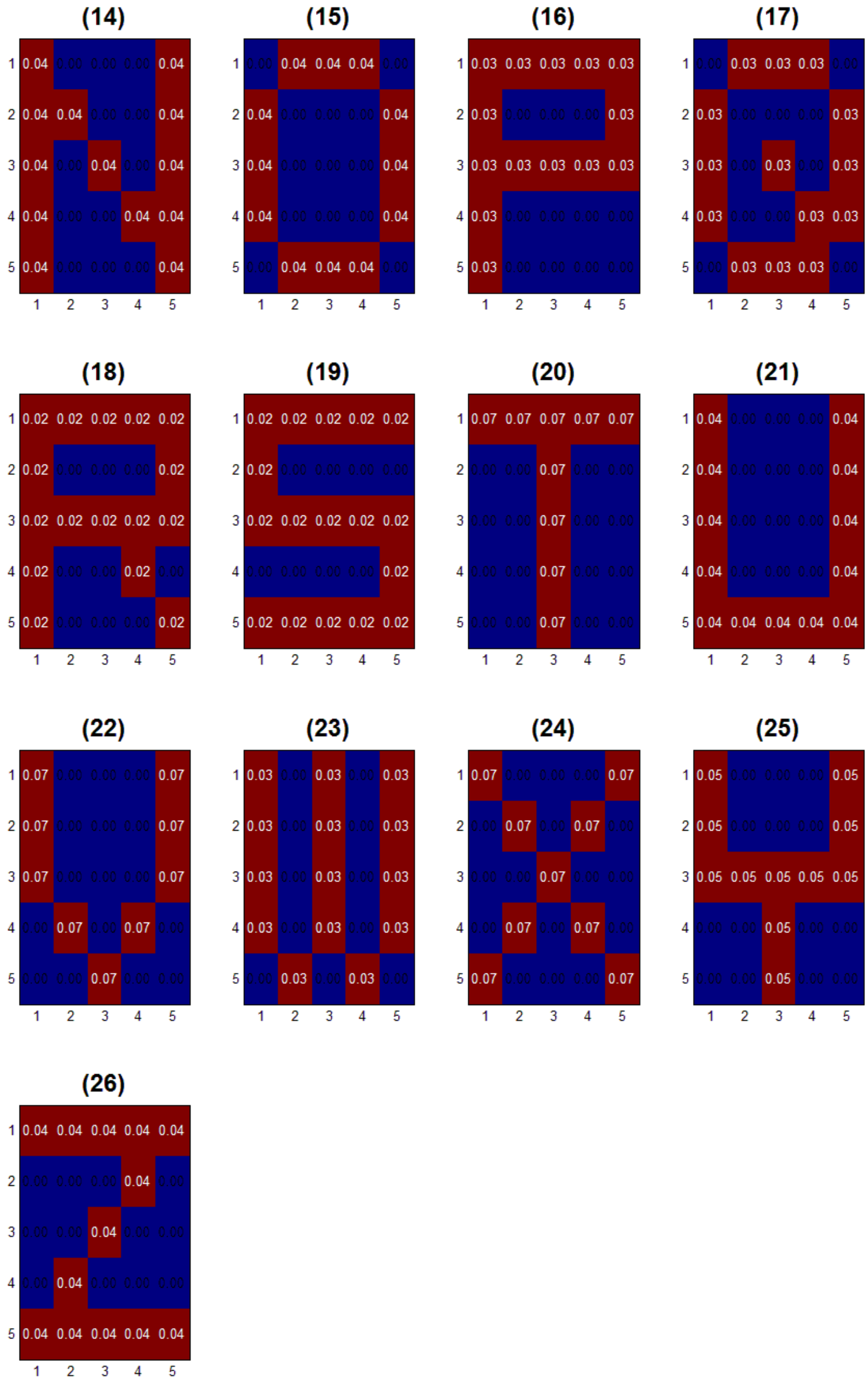

Figure 5.11. Normalized patterns (N to Z) of Fig.5.5.

The network can therefore demonstrate relationship between some patterns and non-relationship for others. But how does it do it? In other words, what is it that the network considers "features" and thereby recognizes a common feature among patterns shown to have a relationship?

Figure 5.10 and 5.11 shows the normalized patterns (of Fig.5.5). They are therefore the normalized input to the network. The figures shows that the patterns H and I determined to have a relationship (Fig.5.8 & 5.7), have the same Hamming weight of 13 non-zero pixels. However, patterns B and R, shown not to have a relationship (Fig.5.9), have different Hamming weights, 17 and 16 respectively. Does this therefore mean that Hamming weights are a measure of feature detected by the network for considering relationships between patterns?

Network output for patterns L and R shows a relationship between them (Fig.5.12). But, they have different number of pixel counts, 7 and 9 respectively. Let us define cumulative pixel value (CPV) to be the sum of normalized pixel values. It was observed that the network considers patterns to have a relationship if their CPV's are equal. Table 5.1 lists the twenty-six patterns and their CPV's. Patterns, J, T, V and X having same non-zero pixel counts (9) were shown to have relationships among themselves. The pattern L having a different non-zero pixel count (7) is also shown to have relationship with the four patterns. We can see from table 5.1 that, these five patterns have the same CPV, 0.63.

The observation was consistent with other patterns. We can therefore conclude that if two patterns have equal CPV's, then a relationship between the patterns is shown by the network regardless of the location of non-zero pixels and number of non-zero pixels.

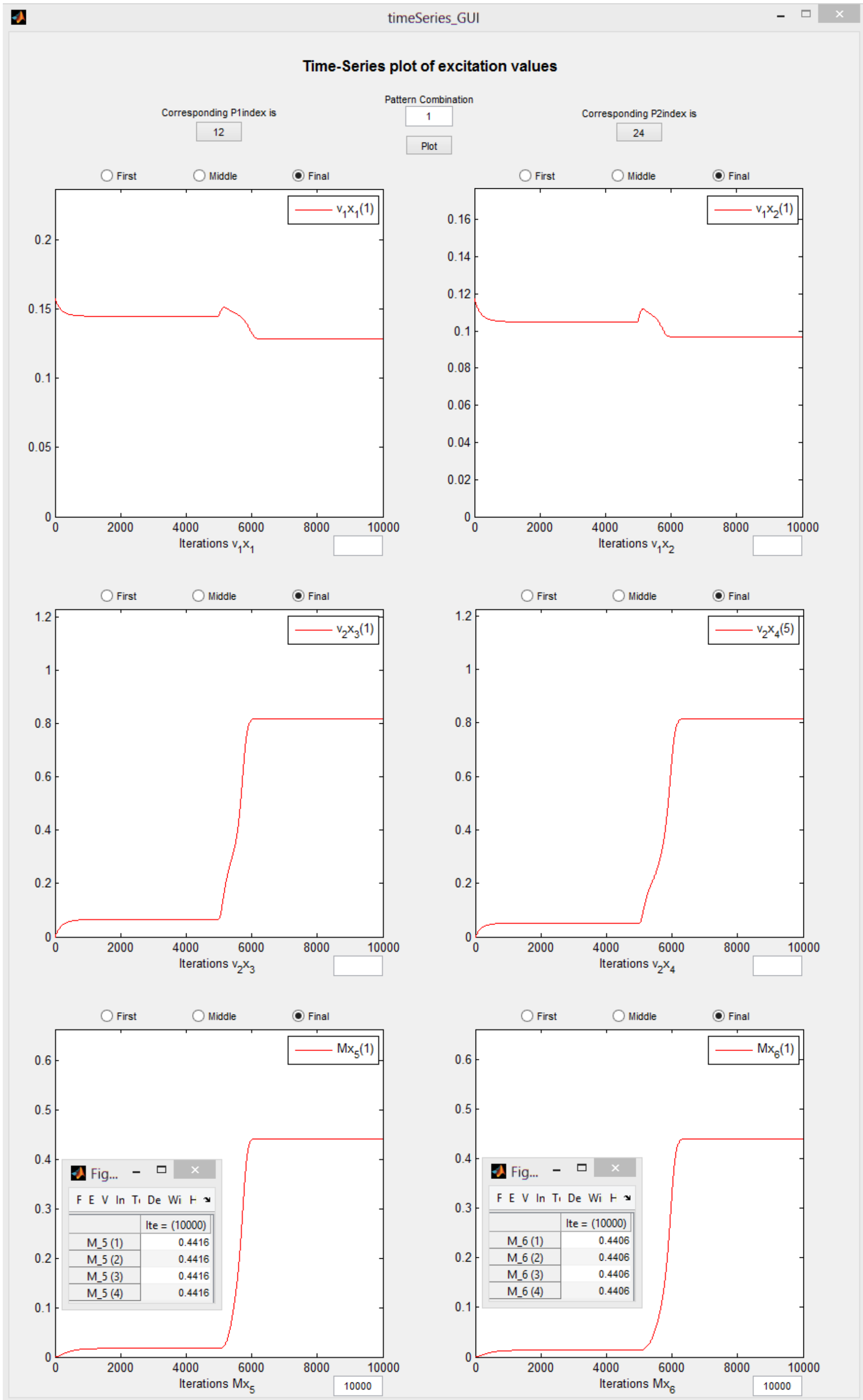

Figure 5.12. Time-series (unscaled) plot for patterns L (left sub-network) and M (right sub-network). This shows the plot of 1st v1 nodes, 1st v2 (winner, left sub-network), 5th v2 (winner, right sub-network) and 1st M nodes. The table in the bottom sub-plots shows the steady-state (10,000 iteration) of the 4 nodes in respective M layer.

Patterns	Number of non zero pixels	Non zero pixel value	Cumulative pixel value
R	16	0.02	0.32
B, E, S	17	0.02	0.34
G	18	0.02	0.36
D, P, Q, W	14	0.03	0.42
A, O	12	0.04	0.48
C, F, H, I, M, N, U, Z	13	0.04	0.52
K, Y	11	0.05	0.55
J, T, V, X	9	0.07	0.63
L	7	0.09	0.63

Table 5.1. Table showing the 26 uppercase English letters arranged with respect to their number of non-zero pixels (out of 25 pixels, 5 x 5 matrix), their normalized value (3^{rd} column) and their total (4^{th} column, CPV or cumulative pixel value). Note that for a particular pattern all its non-zero pixels have the same magnitude of normalized pixel value.

Though CPV is a measure of feature detection by the network, are there other measures? In other words, can two patterns have different CPV but still be shown to have a relationship? The answer is yes.

Let us consider the case such that the left sub-network receives pattern H (Fig.5.8). Then for the right sub-network other patterns with the same CPV (C, F, I, M, N, U, Z) are shown to have relationship with H. But, H (or any other patterns with same CPV) is also shown to have relationship with patterns with different CPV if they both have the same non-zero normalized pixel value (Fig.5.13a). However, the network does not show relationship for patterns (different CPV's) having equal non-pixel value but in the lower end of the normalized scale (Fig.5.13b).

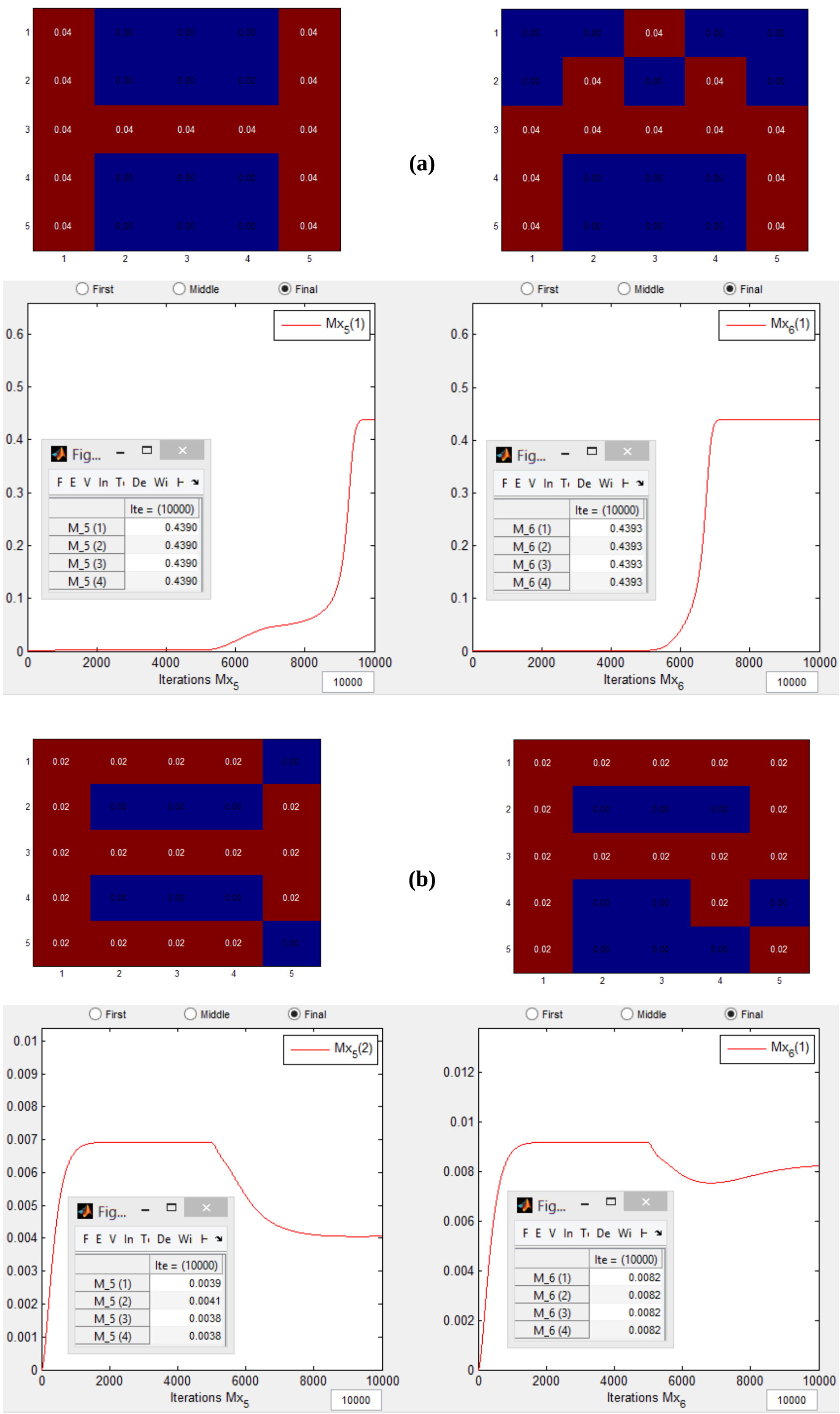

Figure 5.13. Plots (unscaled) showing M layer outputs of patterns H vs. A (a) & B vs. R (b). The table shows that the outputs in (a) fall within a solution set while outputs in (b) differ amongst each other by ~ 2 x.

Therefore, we can conclude from observations consistent with above observations that the network shows relationship between patterns with same CPV. However, if the patterns have different CPV's, the network may still show them to have a relationship if their non-zero normalized pixel value is not at the lower end of the normalized scale.

The results demonstrated above make a novel contribution of significance in ART network theory. The behaviors just described are emergent network properties. Nothing in the design of the networks explicitly inserted the relationship vs. no relationship characteristics just described. This means the theory did not introduce any a priori objective criterion for defining what the network treats as a feature for pattern matching. <u>The experiments just reported are the first-ever demonstration that ART is capable of self-defining feature sets</u>.

The importance of this finding must not be minimized. If a theorist deliberately introduces any objective character of a feature into the design of a network, this amounts to building objective knowledge a priori into the synthesis of apprehension. However doing so is a violation of an epistemological law of mental physics, which holds that human beings are born with no objective knowledge a priori whatsoever. It is therefore a <u>real necessity</u> that any network used in modelling the synthesis of objective perceptions (intuitions) must be capable of <u>self</u>-determining what will or will not constitute an objective feature. This work is the first time any neural network system has demonstrated this capacity, and this is an original contribution to knowledge from this research project.

Above results describes the basic behavior of the network in terms of how pattern-combos either have relationship or not. We must now consider the three properties required for generating equivalence relations, that is, reflexive, symmetric and transitive.

It was observed that every pattern received by the network was shown to be reflexive. In other words, when the left and right sub-networks received the same pattern the M layer output always fell within the solution-set. This behavior is due to the chosen network configuration. Referring back to the model (Fig.5.2d & Fig.5.3), recall that the sub-networks have the same architecture and parameters (Fig.5.6). Since the two similar sub-network are coupled by reciprocal inhibition, it is understandable that the network consistently shows that the relationship between patterns are reflexive. By the same argument, if a relationship is shows between two different patterns then this relationship is always symmetric. This property of the network dynamics is not a hindrance to the act of comparation because generation of equivalence relation is the mathematical act of comparation.

Therefore, if two patterns are shown by the network to have a relationship then the relationship satisfies both reflexive and symmetric properties. These two properties can be therefore be realized from a single pattern-combo. In other words, the network does not have to process another pattern-combo such that they are of same patterns or flipped patterns. If the network had to process other pattern-combos composed of different patterns, it would mean that the realization of the two relation properties would involve a logical ordering process. The network however requires an ordering processing for transitivity. The above minimal neural network does not have this capability.

Comparation is a process in synthesis in sensibility which in turn does not have memory because the obscure parástase from sensory data has not yet become a conscious or objective parástase. The obscure parástase exists however. Therefore, comparation has a 'state'. This means then, if an obscure parástase is a sequence of patterns, the comparation network will be in a 'state' whose consequence would be, a realization (or not) of equivalence relation.

One of the functions of the **pure intuition of time** (**PIT**) within the OB (Fig.3.13) is that it determines 'content' in time. In practice, this means that the **PIT** can determine which pair of elements (in a sequence) is to be processed by the comparation network. Thus we have,

Proposition 9: The comparation network interacts with the pure intuition of time (PIT) such that the PIT determines the order of sequence of pattern pairs to be processed.

Since the aim of the project is to build a comparation network generating equivalence relation and not build other nous processes of the OB, a PIT proxy was built (Fig.5.14). The PIT proxy determines the content for the comparation process by picking pattern-combos one at a time from the pattern sequence. Using results from the above observations that a pattern-combo shown to have a relationship is reflexive and also symmetric, the PIT proxy picks L unique pattern-combos, where L is the length of a pattern sequence.

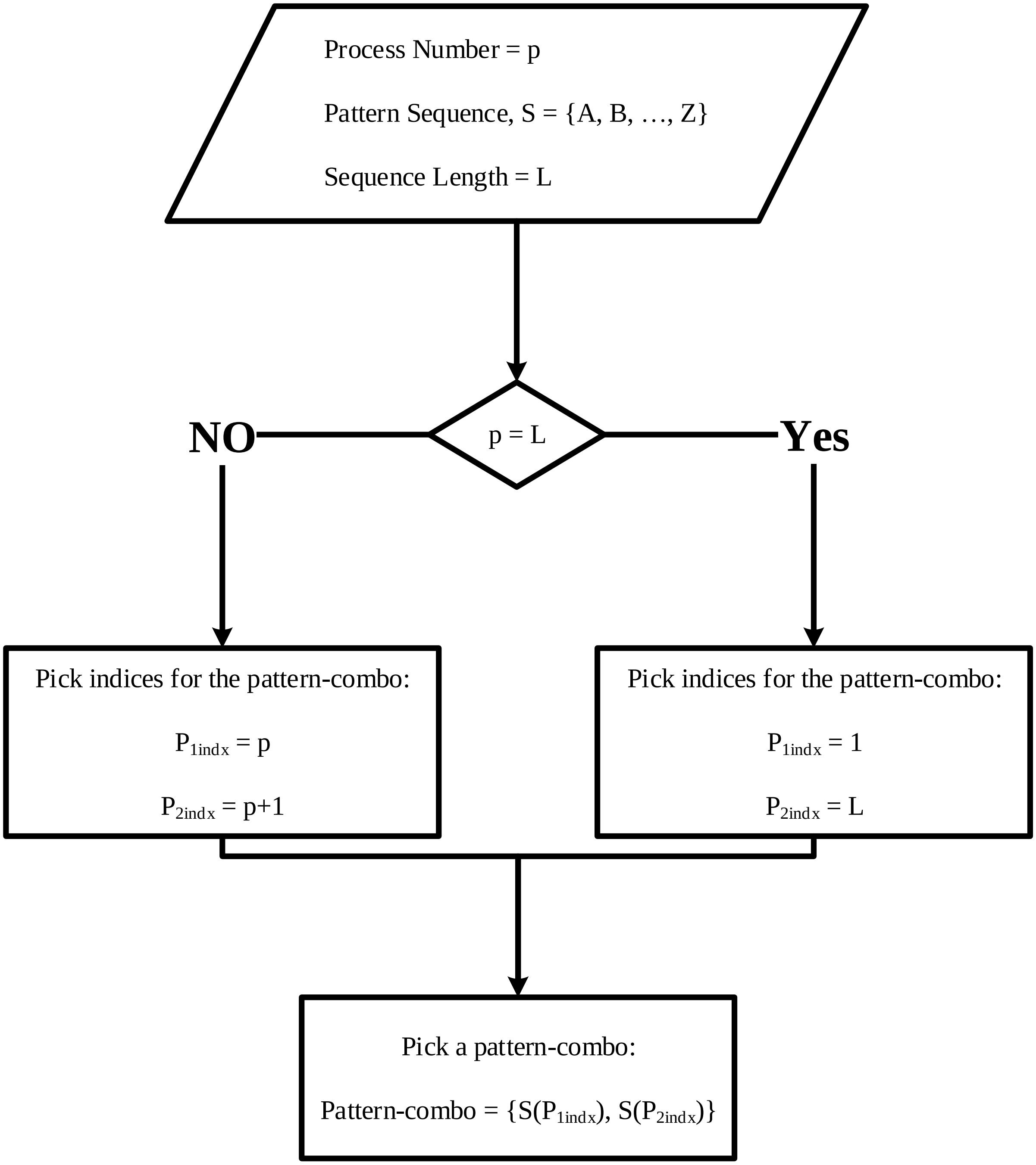

Figure 5.14. The pure intuition of time (PIT) proxy for determining content (pattern-combo) for the process of comparation network (Fig.5.3).

If one of the pattern-combo in the sequence is shown by the comparation network not to have a relationship, then there is no point for comparation to process the remaining pattern-combos. This would mean that the M layer outputs do not fall inside the solution-set. In other words, the M layer outputs are not in equilibrium and hence not expedient.

Since the PIT is a sub-process within the synthesis in sensibility (Fig.3.11 & Fig.3.13), like comparation the PIT is also linked to the reflective judgment (TRJ). Thus, the TRJ judge expediency based on the two pre-motor images and then stops the comparation process if one of the pattern-combo in the pattern sequence is not expedient. However, TRJ does not have the capability to directly stop comparation but it can make PIT be known that a pattern-combo is not expedient. Hence,

Proposition 10: The function of PIT determining the content of the comparation process can short-circuit the order of pattern-combos based on non-expediency judged by TRJ.

The judging of M layer outputs for expediency was performed by a TRJ proxy for this particular function. Since Weber and Fechner, the concept of 'just noticeable difference' has been well known psychological phenomenon [Kendel, 2000]. Based upon this notion, the TRJ proxy was built by comparing the arithmetic difference of steady state (last iteration) M-layer outputs against a parameter, δ akin to difference limen or just notifiable difference. Figure 5.15 shows the new minimal network model based upon the previous minimal anatomy (Fig.5.2d) incorporated with the PIT and TRJ proxies. Notice that this function of the reflective judgment is not exactly the same as judging expediency of the comparation network which was incorporated within the model (Fig.5.2d & Fig.5.3).

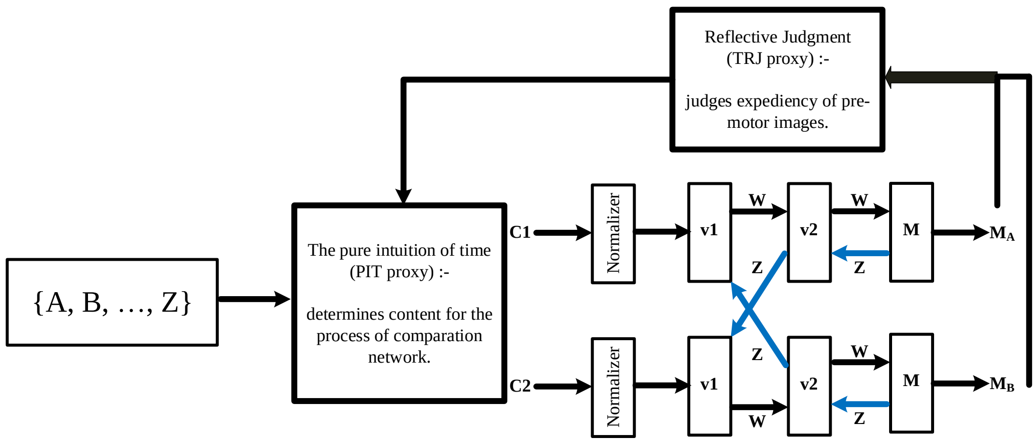

Figure 5.15. The minimal anatomy for generating equivalence relation. Notice that this is based on the previous minimal network (Fig.5.2d) with the addition of PIT proxy (Fig.5.14) determining the content (comparands, C1 & C2) from a sequence of patterns and also receives report of expediency from TRJ proxy.

Behavior of the proposed minimal anatomy.

The parameters used were the same set as earlier (Fig.5.6) with the addition of $\delta = 10^{-3}$, for the TRJ proxy. The patterns comprising any desired sequence was picked from the same set of upper-case English alphabets (Fig.5.5). Because normalizer parameters are the same, the resulting normalized inputs are also unchanged (Fig.5.10 & 5.11).

The minimal network showed equivalence relationship for some pattern sets and not for others. For a pattern sequence {C, F, A, O}, the network process finds relationship between C & F (C α F) and so it continues for {F, A}, {A, O} and finally {C, O} (Fig.5.16). On the other hand, for {C, A, D, G}, the network finds C α A and proceeds to {A, D} but finds A ~~α~~ D and hence stops the process (Fig.5.17a). This short-circuiting of the process is also seen in pattern sequence {B, D, E, P, S, G}, where the process stops at {B, D} (Fig.5.17b). Notice that, if the pattern sequence was composed only of either {B, E, S} or {D, P} the network would continue the process and find relationship among the respective patterns (Fig.5.18).

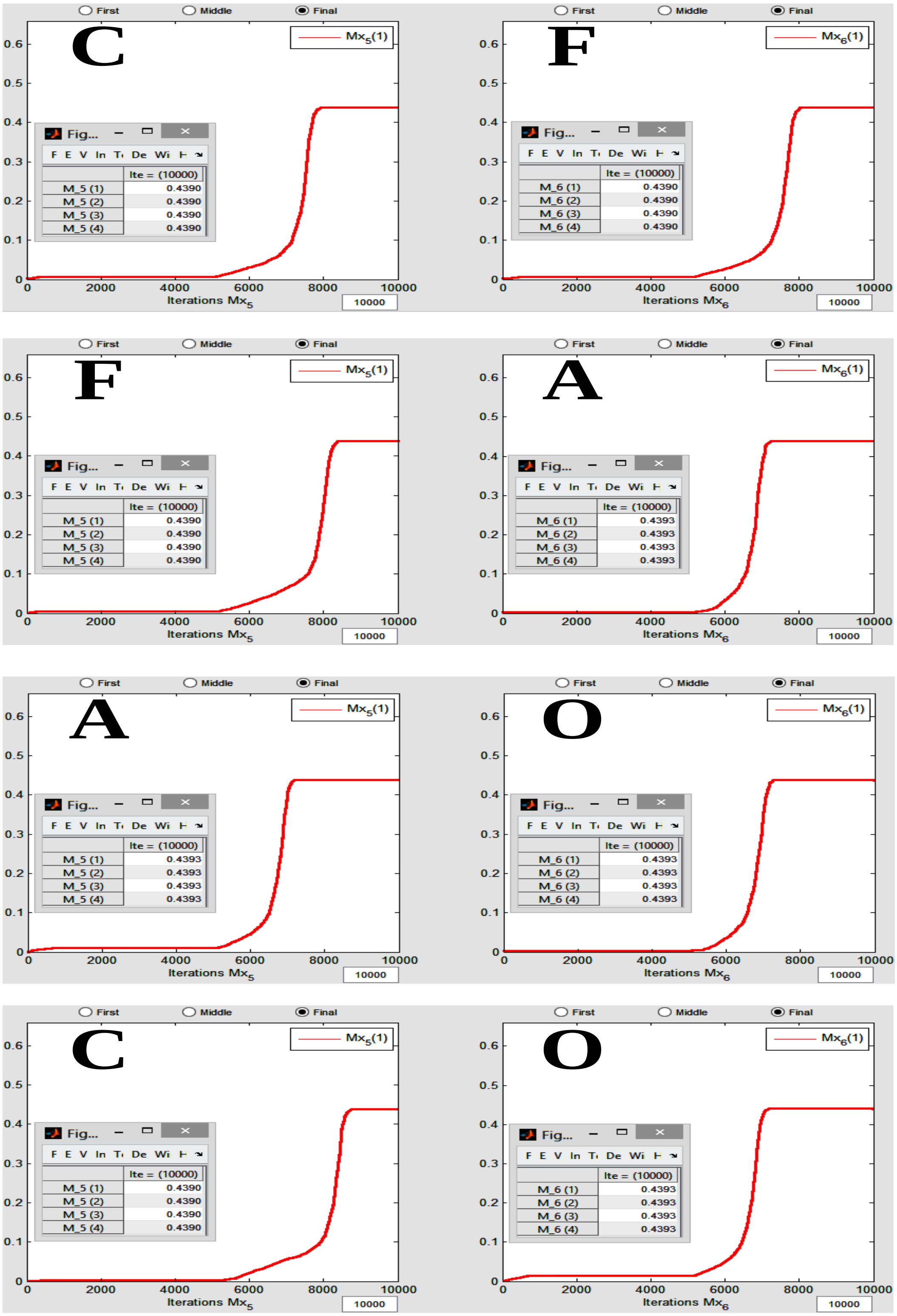

Figure 5.16. Plots (unscaled) showing M layer outputs for respective pattern-combos in the pattern sequence {C, F, A, O}. From the top, the network proceeds from {C, F} and then {F, A}, {A, O} and finally {C, O}. Here all the patterns are shown to have a relationship.

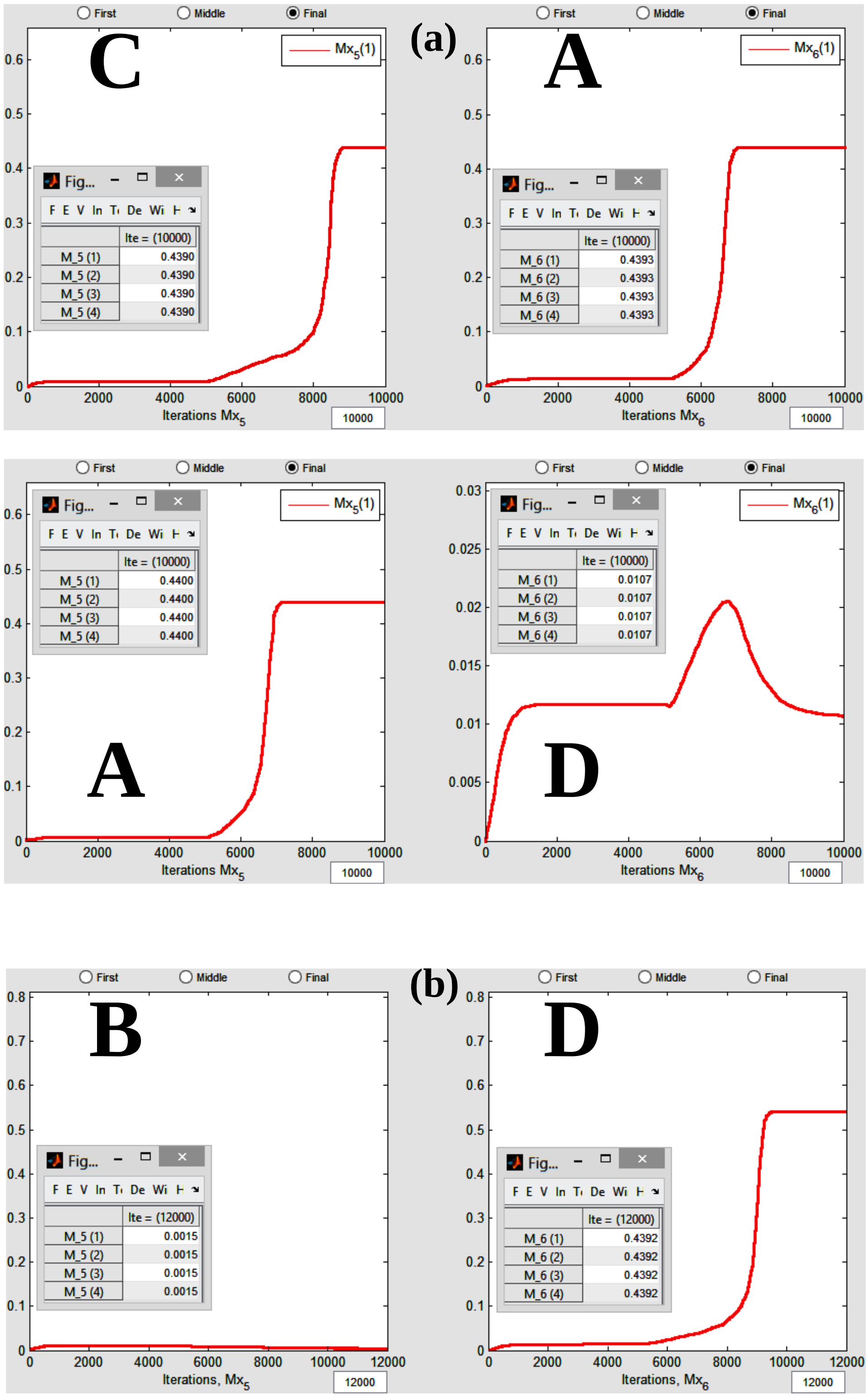

Figure 5.17. Plots (unscaled for (a), scaled for (b)) showing M layer outputs for respective pattern-combos in pattern sequence {C, A, D, G} for (a) and {B, D, E P, S, G} for (b). In (a) the network finds relationship in {C, A} and {A, D} but not in {D, G} and hence stopped for {C, G}. In (b) the network stops after {B, D}.

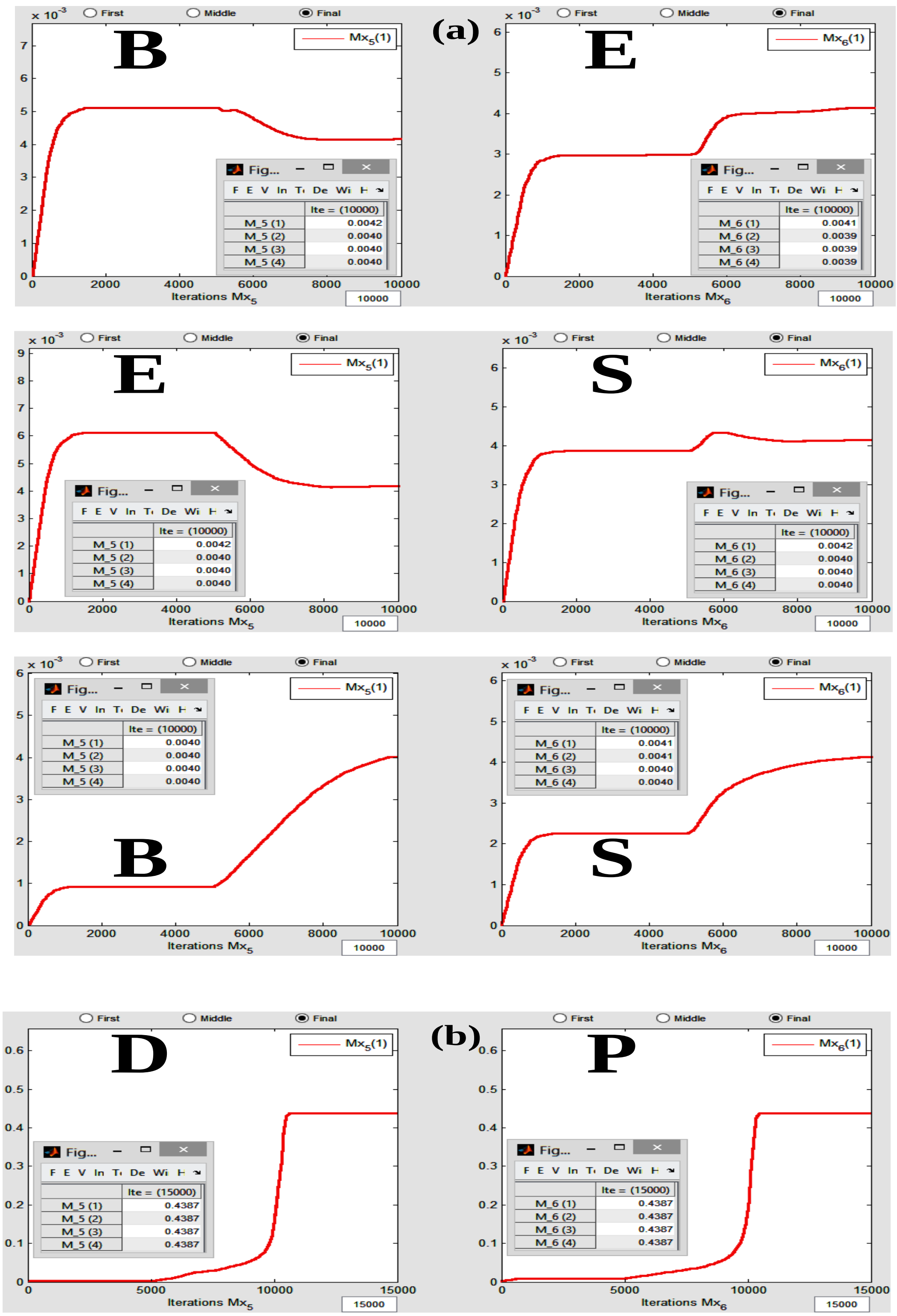

Figure 5.18. Plots (unscaled) showing M layer outputs for respective pattern-combos in pattern sequence {B, E, S} for (a) and {D, P} for (b). Notice that the patterns in both these sequences are the same as the sequence for Fig.5.17b. In this case however, patterns in both the sequences are shown to have a relationship.

These demonstrations do not strictly show the complete synthesis of an equivalence relation. To do that transitivity would have to be demonstrated over the whole of the candidate set of patterns. For instance, demonstrating transitivity for the set {C, F, O, A} requires, in addition to the demonstration of figure 5.16, that the following pairs have the same relationship:

C → O,
F → A.

However, recall that if a relationship is found between patterns then as a dynamic property of the network, the relationship is reflexive and also symmetric. Thus for the pairs that have been shown to have the same relationship (Fig.5.16), due to the property of the network dynamics, the following pairs will also have the same relationship:

C ← F (since, C → F),
F ← O (since, F → O),
O ← A (since, A → O), and
C ← A (since, C → A).

In other words, the relationship in the pattern-combos implies that the patterns are interchangeable, i.e., C & F are interchangeable and F & O are interchangeable. Thus, the following pairs will have the same relationship:

C → O (since, F ← O), and
F → A (since, C → F),

Simulations (figure not shown) of these pattern pairs confirm this.

The demonstrations provided here shows that the generation of equivalence relation by means of the minimal network of figure 5.15 is possible, and this is sufficient to demonstrate the Verstandes Actus of comparation. In this, the particular matter of equivalence is irrelevant, as it must be according to the laws of mental physics.

In the OB, determination of the process of the PIT is regulated by ratio-expression from practical Reason acting through determining judgment (Fig.3.11). Therefore, the complete synthesis of objective equivalence is realized by means of the overall synthesis of judgmentation.

In conclusion, the above minimal neural network (Fig.5.15) has the ability to generate equivalence relations. With the exception of pattern sequence length L = 2, if the network demonstrates relationship for the L^{th} process then the patterns within that sequence are candidates for an equivalence relation. It should be emphasized that the generation of equivalence relations is based on network dynamics without any ad hoc knowledge. In other words, the relationships are generated without introducing any a priori information about the patterns.