Nonlinear Dynamics and Systems Theory, **3**(1) (2003) 43–63

Generic Emergence of Cognitive Behaviour in Self-Generating Neural Networks

C.S. Bohun[1], S.J. Carruthers[2], R. Edwards[3] and R. Illner[3]

[1] *Department of Mathematics, Penn State Mont Alto,*
1 Campus Drive, Mont Alto, PA, USA
[2] *Department of Mathematics & Statistics, Simon Fraser University,*
8888 University Dr., Burnaby, B.C., Canada V5A 1S6
[3] *Department of Mathematics & Statistics, University of Victoria,*
P.O. Box 3045, Victoria, B.C., Canada V8W 3P4

Received: May 3, 2002; Revised: December 13, 2002

Abstract: We discuss the design and behaviour of families of neural networks which grow out of a small set of “mother” neurons in response to external stimuli and to the activities present in various parts of the net at a given time. The growth process is subject to a few fundamental rules, like

- the ability of neurons to grow new neurons or connections is gradually exhausted with the number of generations
- neurons are either of excitatory or inhibitive type
- inhibitive neurons have a tendency to form long-range connections, whereas excitatory neurons “prefer” short-range connections.

In addition, there are a number of free parameters in the equations driving the time evolution of the neural activities. Our design is implemented using Matlab, such that the growth process of the network and its activity can be observed and controlled interactively on the computer screen.

Once the networks are grown both periodic attractors and fixed points are observed generically in response to external input. The inputs used in the network’s formation are typically distinguished by characteristic responses, but the resulting networks are capable of other behaviour in response to other inputs.

Keywords: *Neural networks; growth rules.*

Mathematics Subject Classification (2000): 92B20, 68T05, 37N25.

1 Introduction

Standard artificial neural networks are either feed-forward nets of perceptron type, or they allow feedback between the various layers of neurons. The latter type is usually described as a dynamical system, and the equations for the membrane potential (or firing rate, or activity) of the individual neurons form a coupled system of nonlinear differential equations. For surveys on the theory of these nets, we refer the reader to the books by Arbib [2], Hertz, *et al.* [11], Amit [1], or the (older) papers by Grossberg [8] or Lippmann [12].

What all these nets have in common is that their graph structure is fixed once and for all. Connections exist or not, and training or learning of the net is done by adjusting the connection weights. This imitates the biological process of changing synaptic efficacies; various learning rules are used, the most common being variants of the Hebb Rule ([9]). In essence, a neural net consists of the following three elements:

1. A set of neurons at certain locations; each neuron can send output or receive input via connections. A connection is a one-way street; from the point of view of the sending neuron, we will call it an axonal (output) connection; from the point of view of the receiving neuron, we will call it a synaptic (input) connection. In simple additive networks, the rate of change of a neuron's membrane potential is determined by a sum of synaptic inputs, each of which is a nonlinear function of a presynaptic neuron's membrane potential (which may be interpreted as a firing rate), modulated by a "synaptic efficacy" (connection strength).
2. A set of existing connections between the various neurons, which gives the whole object the structure of a directed graph.
3. A matrix of connectivity (synaptic) strengths, which can change in the learning process.

In these standard nets, the first two of these elements are fixed; only the third is subject to change. Because of this, the traditional method of developing an artificial network is to connect every neuron to every other neuron with some initial connection weight (if the network is layered then typically only neighbouring layers are fully connected) and then to evolve the connection weights according to some training process. For nets that consist of a large number of neurons, this training process can become quite awkward.

In any case, biological evidence points in a different direction. Complicated biological neural networks, i.e., brains and nervous systems, evolved from simpler ones in an evolutionary process, a process which is repeated following a genetic blueprint with some accuracy in each individual as the brain develops. The end results are large brains, with as many as billions of neurons but a relatively sparse connection matrix. Even small subsystems of the brain are far from fully connected. These natural neural networks display a wide variety of principles of organization ([14]) including "layered" structures (e.g., separation between "processing" neurons and data gathering neurons, with relatively few intermediate layers, cortical layers and columns, or different 'nuclei' or structures in the brain), interplay between excitation and inhibition (sometimes "short-range excitation, long-range inhibition" to produce "cell assemblies", sometimes pairs with negative feedback, such as motor neurons and Renshaw cells), and dynamic responses (e.g., oscillations) to inputs.

Neurons differ from other body cells in the sense that they do not split. As a consequence, a mature brain is a more stable biological structure than, say, a muscle or any other organ, where the individual cells are not so important. Neurons, once they die, can possibly be replaced, but the functionality of the brain is mostly maintained by the sharing of tasks by many neurons. This redundancy is indeed a central feature of biological neural nets.

It is easy to think of reasons why neurons in the mature brain might not have been designed to split. Brain functions are more complex than the functions of other organs, and the way that neurons interact is crucial for this function. Neural cell division on a large scale would presumably alter brain function in ways incompatible with maintaining the multiple specific tasks assigned to the brain by nature.

In the process of evolution, however, such splitting must have occurred in one way or the other, because otherwise there would be no sophisticated brains at all (when we use the word "brain" here, we mean any biological neural structure with at least a few dozen interconnected neurons). Splitting of some kind must also occur in the growing brain, albeit controlled by the genetic code.

The objective of the project described in this article is to attempt a simulation of an evolutionary growth processes for neural networks. Although not truly evolution, the process used has been inspired by the biological idea. Our purpose is to develop a network displaying characteristic responses to inputs used to guide the growth. We start with a network of very simple type, namely a few neurons which are present from the outset and whose activity is interpreted as the information processing done by the emerging network (we call these neurons the "mother neurons"). The mother neurons have the ability to split, or, more accurately in our implementation, "sprout" new neurons, which can themselves sprout again, or alternatively, form connections to already existing neurons. The resulting "child" neurons, "grandchild" neurons etc. are connected to their parent by a synaptic connection, and they can, with some randomness in the process, grow axonal connections to other neurons and "share" their activity, as measured by their membrane potentials. The whole process is driven by external inputs and by the activity levels of the existing neurons at any given time. The decision as to when and where to connect or sprout is deterministic. However, when the "best" receiving location is not unique, one is chosen randomly. Synaptic connection strengths change over time and can be excitatory or inhibitive, but a given neuron can grow either excitatory or inhibitive connections, not both. We shall therefore classify neurons as either excitatory or inhibitive, depending on what type of connections they grow. Inhibitive neurons are never allowed to sprout, because the activity of a "child" of such a neuron would be immediately suppressed and therefore would contribute little to the information processing ability of the net. Instead, when an inhibitive neuron grows a connection, this connection must lead to an already existing neuron.

We implemented limits on the size of the net by setting low upper limits on how many axonal connections to new or existing neurons a given neuron can form and how many synaptic connections it can receive. For all the details of the process, see Sections 2 and 3.

The growth process of the neural networks presented here is dependent on the current spatial configuration. As a courtesy to the human visual system, the system evolves on a two-dimensional grid, and the direction in which a new neuron sprouts from an existing one depends on a "potential" created by the neurons already in place. Neurons sprout downward from a top layer (occupied by the mother neurons), and the generation number of a neuron is identical to the number of its layer. In this sense, all of the mother neurons are first generation neurons. The potential has the effect of allowing neurons to "feel" each others presence even without being connected. As well as depending on the location and type of the neurons in the network, the potential is also dependent upon both the number of available synaptic connections left in other neurons and the generation number of each neuron. This latter dependence was introduced after a number of numerical experiments indicated that a sufficient number of inhibitive back-connections are required for the network to exhibit interesting behaviour. Depending upon the neuron-type (excitatory vs. inhibitive) the neuron can either sprout or connect. Recall that inhibitive neurons can only connect. Rather than searching the complete two-dimensional grid for viable locations, the sprouting/connecting is only allowed inside a certain window whose size depends on the generation (level) and on the neuron-type. Connections are allowed anywhere inside this window but sprouting is further restricted by allowing each neuron to only sprout down one level. Inhibitive neurons differ from excitatory neurons in the sense that they connect preferably over longer distances, and preferably to neurons of earlier generations.

We dwell further on the problem of why the mathematical analysis of complex neural networks seems so hard if attempted directly. A network of, say, 100 neurons, in which each neuron is connected to 10 others, will be described by 100 coupled nonlinear ordinary differential equations, with 10 coupling terms on the right hand side of each equation. If equations for the connectivities (on a slower time scale) are added, the number of equations will grow by another thousand. Even if we disregard these latter equations for the modeling of the neural dynamics, the state space of the system has 100 dimensions, and it seems nearly hopeless to predict the dynamical behaviour of such a system in the general case without some structural principle, or its response to external inputs (modeled as force terms on the right-hand sides of the equations). Even for much simpler systems of differential equations, with only 3 or 4 dependent variables, prediction of the dynamic response to inputs is a real challenge (for a case study on the Lorenz system of equations, see Evans, *et al.* [7]). Attempts to simplify the description of such systems by nonlinear diffusion equations were made by Cottet [4] and, in greater generality, by Edwards [5], but the analysis done in these papers shows that such mean field approximations are only feasible under very special assumptions on the connection matrix (in Edwards [5]), connections are local, predominantly excitatory, and nearby connections cannot have very different strengths). Other structural principles, such as symmetry of connections (Hopfield [10]), tend to be overly restrictive and un-biological.

Yet nature has found ways to design much larger networks, described by systems of equations of unimaginable complexity, which display highly complex yet highly organized behaviour. Our paper is an attempt to repeat such an organic design process in a computer.

2 The Model

We now describe, in a series of steps, how our network starts from the mother neurons and grows in response to external stimuli and, eventually, internal activity. The activity of a given neuron is identified with the membrane potential of that neuron. External inputs are identified with force terms on the right-hand sides of the equations. For a given size of the net (say n neurons), the evolution of the membrane potentials with respect to time is given by network equations of the form

$$\dot{u}_i = -u_i + \sum_{\substack{j=1 \\ j \neq i}}^{n} T_{ij} \ \mathrm{sgm}(\mu(u_j - w_j)) + I_i, \qquad i = 1, 2, \ldots, n, \tag{2.1}$$

where u_i is the membrane potential of the i-th neuron. The three terms on the right hand side are respectively: a leakage term, the contribution from all the neurons connected to the i-th neuron and an external force term. The connection strength between the neurons with label j and i is denoted as T_{ij}, where j is the label of the sender and i is the label of the receiver. The external force term is denoted as I_i and is interpreted as a temporary input upon neuron i. This can be switched on or off, and can take any real value inside an interval $[-I_{\max}, I_{\max}]$, thereby enhancing or inhibiting the activity of the i-th neuron. The symbol sgm denotes a sigmoid function and for this paper, we take

$$\operatorname{sgm}(x) = \frac{1}{2}\,(1 + \tanh x) = \frac{1}{1 + e^{-2x}}, \tag{2.2}$$

which assumes values in (0,1). Other sigmoids are certainly possible. The constants w_j are inserted as firing thresholds which further differentiate the activity of inhibitive from that of excitatory neurons. In our experiments, we chose $w_j = 0$ for all excitatory neurons, $w_j = 0.5$ for all inhibitive neurons. The rationale for this, in a discrete time context, is explained in [6] and [13]. In essence, by first choosing the constant μ sufficiently large and then combining inhibitive neurons with a positive threshold with excitatory neurons with threshold at zero creates conditions that are likely to lead to oscillatory and perhaps even chaotic behaviour of the net in the absence of external input. The experiments discussed in Section 4 confirm this. The constant μ is known as the gain, and

$$v_i = \operatorname{sgm}(\mu(u_i - w_i)) \tag{2.3}$$

is the firing rate of the ith neuron. By construction, the membrane potential takes values in $\mathbb{R}$, whereas the firing rate takes only values in (0,1).

We will also need a rule to initialize the connection strengths T_{ij}. Once a connection strength is determined, we could apply a learning rule to adjust it dynamically, but we chose to leave it fixed for the lifetime of the network so that effects of the growth process would not be confounded by subsequent learning effects.

The basic plan behind the development of the network can be encapsulated with a few simple rules.

1) Neurons are grown in layers.
2) Excitatory neurons form in locations of low activity.
3) Inhibitive neurons form in locations of high activity.
4) Input that changes in time modulates the activity.
5) Input is distributed on the x-axis, while layers are formed in the y direction.

Because of this design plan, the network is grown on an $M \times M$ grid in response to a series of inputs which are visualized as being applied from the bottom. The position of a neuron is therefore given by a pair of numbers (a, b), $a \in 1, \dots, M$, $b \in 1, \dots, M$. The horizontal coordinate a is cyclic in that the position $(a + kM, b)$ is identified with (a, b) for all $k \in \mathbb{Z}$. The vertical coordinate is grouped into a set of equally spaced disjoint intervals the size and number of which depends on the size of the grid, M, and the maximum number of neurons that could be placed. These vertical intervals are referred to as layers, levels or even generations. If we denote the width of a level as Δl and the width of the gap between levels as Δg then level $k+1$ is defined as the points (a, b) with

$$M - k(\Delta g + \Delta l) \le b < M - k(\Delta g + \Delta l) + \Delta l.$$

In the special case of level one, $k = 0$, we take $b = M$. One additional (arbitrary) constraint is that the bottom of the last layer must lie at $b = 2$. This implies that the number of layers is $1 + (M-2)/(\Delta g + \Delta l)$. For the numerical experiments we chose $M = 30$ and because of the maximum number of expected neurons, the vertical position is partitioned into five disjoint subsets each with a width of $\Delta l = 3$ grid points and separated by a gap of $\Delta g = 4$ grid points. This is illustrated in Figure 2.1. To start the net, we place N mother neurons equidistant from each other on level one. In the case of $N = 3$ these are placed at (5,30), (15,30), (25,30). The growth process is then driven by inputs located in the lowest layer. When the system is first defined, no neurons exist at this input layer. Therefore during the training process each neuron experiences a weighted average of the signal applied at the input layer according to the expression

$$I_i = \frac{1}{n_i}\sum_{j=1}^{n_i} f(x_j), \qquad x_j = x_i - \frac{n_i-1}{2}, \dots, x_i + \frac{n_i-1}{2}.$$

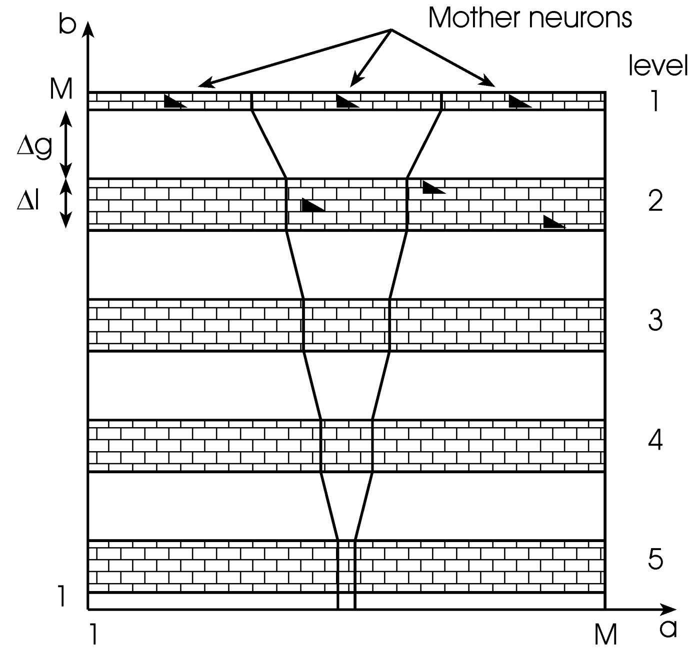

Figure 2.1. Shown here is the location of the various layers of the network used in the numerical experiments. It depicts the location of the mother neurons, the various levels and the location of the input layer. Also included is the vertical cone extending from an external input located at $(a, b) = (15, 1)$.

Here I_i is the external force that appears in equation (2.1) and is applied to neuron i by the input vector f. The number of points over which to average, n_i, is governed by the layer of the i-th neuron and in the case of our 30×30 array we chose $\{n_i\}_{i=1}^{5} = 11$, 7, 5, 3, 1 for layers 1 thru 5 respectively. Consequently, the inputs applied at the bottom layer are *felt* by all of the neurons that lie inside a vertical cone that emanates from the location where the input is applied. Once again, refer to Figure 2.1. After the training process, the input level for all of the layers is divided by a factor $5 - l + 1$, where l is the layer of the neuron under consideration. Ultimately, this means that the external input I to the mother neurons is suppressed by a factor of one over the maximum number of layers while the neurons at the bottom of the network, $l = 5$, feel the external input directly.

3 The Growth Process

3.1 The mother neurons

The mother neurons are really just ordinary neurons, but they have the ability to sprout new neurons, to which they will be connected via axonal connections, in response to their own activity. Specifically, the mother neurons can initially receive an input, i.e., be stimulated externally from the bottom, and if their activities rise above a predetermined threshold τ, they will sprout an axon with a child neuron. The mother neurons are by definition excitatory neurons, meaning that the axonal connection which connects them with children is excitatory. The child neurons can be either inhibitive or excitatory; the choice is made randomly, and the probability that a sprouted neuron is inhibitive is a free parameter in the program, denoted by ρ. We varied ρ from layer to layer, starting with $\rho = 0.2$ for the second layer and rising to $\rho = 0.6$ for the bottom layer.

If a child neuron is excitatory, it has again the ability to sprout new neurons in response to sufficient activity. Recall that the sprouting takes place when the *activity* is above the threshold; for a sufficiently large network, this can happen even when the neuron receives no external input. If the child is inhibitive, we do not allow it to sprout "grandchildren"; its only way to build connections is to connect back to the mother neuron or, later, to other neurons already in existence.

We require that the mother neurons can sprout child neurons, or connect via axonal connections to other neurons (e.g., grandchildren), only a finite number of times; the mechanism which we use to enforce this is to define an integer G which is the maximal possible number of generations that can sprout from the mother neurons. So that we generate a nontrivial network, we require that $G > 1$. The total number of axonal connections for any neuron is $G+1$ so that even if a mother neuron has sprouted G times, there is still one axonal connection available to connect to an already existing neuron. For the next generation, only $G-1$ of the total $G+1$ potential axonal connections are allowed to sprout, etc. This rule implies that every subsequent generation can have one fewer sprouted connection than the previous generation and is a convenient (and natural) way to limit the size of the net. Indeed for our 30×30 array we have five layers so that $G = 4$ and therefore using the above conditions, the network can have a maximum of 195 neurons. Since $\rho \neq 0$, we usually have inhibitive neurons and as a result the expected number of total neurons is less.

If g_i is the generation label of the i-th neuron, then g_i is an integer between 1 and $G+1$. By $a_i(t)$ we denote the number of remaining potential axonal connections of the i-th neuron at time t. If t_* is the moment of creation of this neuron, then $a_i(t_*) = G+1$ and the number of these potential connections that can sprout is $G - g_i + 1$.

The third characteristic number associated with each mother neuron is again an integer, which we denote by S, the maximal number of synaptic connections which the mother neuron (or any other neuron) can potentially accommodate. We choose $S = G$.

We summarize: In the beginning, we create N mother neurons which are characterized by three parameters, τ (threshold for sprouting or connecting), G, the number of admissible axons that can sprout from the mother, and S, the number of possible synaptic

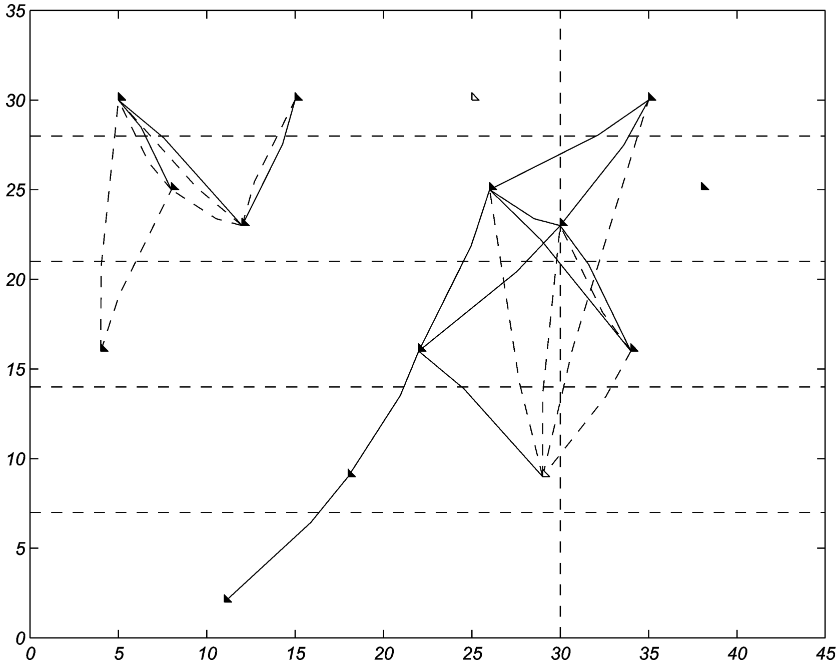

Figure 3.1. Illustrated above is a sample network showing a few mother neurons along with their children. When a new neuron is created in the i-th layer (generation $g_i = i$) it is assigned $a_i = G + 1$ potential axonal connections ($G + 1 - g_i$ can sprout) and S potential synaptic connections. As the network evolves both of these values can decrease. The horizontal dashed lines divide the net into the five levels while the vertical dashed line indicates, where the net wraps around so that the neuron at (35,30) is another image of the neuron at (5,30). Solid connections are excitatory; dashed connections are inhibitive.

connections. Both S and τ are the same for all neurons while the number of admissible axonal connections that can sprout depends on the layer in which a particular neuron finds itself. Figure 3.1 gives a simple network that has begun to evolve under these assumptions.

3.2 The next generations

The neurons of the next generation (the children of the mother neurons) are set to have essentially the same properties as the mother neurons: They have again room for S synaptic connections, and their threshold for sprouting or connecting is again τ. However, we allow them to sprout only $G - 1$ times, and the next generation will be allowed to sprout only $G - 2$ times, etc. This is a built-in mechanism to limit the size of the net, and it follows that the net can never grow to more than

$$N_G = N \sum_{k=0}^{G} \frac{G!}{(G-k)!} = N \sum_{k=0}^{G} \binom{G}{k} k!$$

neurons, where N is the number of mother neurons (which are also thought of as the information processing neurons). For the situation $N = 3$ mother neurons and $G = 4$, there is a total of $N_G = 195$ possible neurons.

Once a few neurons are in place, neurons which receive an input above the chosen threshold do not necessarily have to sprout a new neuron. We also give them the option to build a connection to an already existing neuron, e.g., to a mother neuron, a sister neuron, or any other neuron with room for synaptic connections. In fact, as already mentioned above for the first child of the mother neuron, inhibitive neurons have only this option.

We have so far only briefly discussed the geometric structure of the emerging net. In fact, it is conceivable to ignore this aspect completely and simply not assign the new neurons a spatial location at all; in such a model, every neuron will consider every other one as a candidate for a possible connection, and may make a random choice among these candidates. Numerical and graphical experiments with such a setup proved messy and showed that it was all but impossible to explore interesting correlations between the given input, the structure of the net and the output.

Therefore, we postulated that our neurons actually had to have a spatial location as described above. As a simple mechanism to record which sites in the array are occupied and by which type of neuron, we introduce two matrices. The first is an occupation matrix O_{ij} whose entries are 0 at vacant sites and are the labels of the neurons at occupied sites (neurons are labeled in the order in which they are created). The second matrix Z_{ij} encodes the neuron type and has entries 1, -1, or 0, depending on whether the site is occupied by an excitatory or inhibitive neuron, or vacant. These two matrices could be easily combined into a single matrix.

We further postulate that by their mere presence, the neurons create a potential field which determines the direction of new axonal connections. Such connections may terminate in new neurons (sprouting) or at neurons already present (connecting). Inhibitive neurons have only the second choice. The potential is a function of the state of the system at a given time. We define this state by the neurons which exist at this time and their properties. The neuron with label k at a location (site) $(x_k, y_k) \in \mathbb{Z}^2$ in level g_k of type Z_{x_k,y_k} has activity (membrane potential) $u_k(t)$, $s_k(t) \leq S$ free synapses and $a_k(t) \leq G + 1$ free axons. We sometimes use the shorthand $Z_k = Z_{x_k,y_k}$.

The state of the system at time t, when K neurons have been created, is then given by the set of coordinates $\{(x_i, y_i, g_i, s_i(t), a_i(t), Z_i);\ i = 1, \dots, K\}$. Notice that neither the membrane potential, nor the number of existing connections play a part in the definition of current state.

Modifications of our model could certainly include these other characteristics into the definition of state. However, the current potential depends only on the variables defined above. We tried various candidates for potentials; the one which seemed to lead to satisfactory patterns is the following:

$$V(x,y) = \sum_{i=1}^{N} \frac{S/2 - s_i + \gamma|G - g_i|}{1 + |x - x_i| + |y - y_i|}, \tag{3.1}$$

where γ is a free parameter. We found $\gamma = 1/2$ to work quite well for our numerical experiments. There are two effects that are modelled in the potential. The $S/2 - s_i$ term has the effect that a neuron at site (x_i, y_i) will make a negative contribution to the potential while it has many free synapses; as its number of free synapses, s_i, decreases, this potential will gradually increase and eventually become positive. The $|G - g_i|$ term makes neurons which are vertically far away more attractive. We will see below that this tends to create longer back connections.

If our potential were a potential in the usual physical sense, forces between various neurons would act in the direction of the gradient of the potential. We loosely follow this idea in using the potential to give directions for sprouting or connecting. To do this, we need one more ingredient, namely the idea that neurons will look for sites to sprout to, or connection partners, in a neighborhood rather than far away. We implicitly work under the assumption that there is a natural scale of length for the typical connection, i.e., that the axonal connection cannot exceed a certain length. This length defines a "window" around each neuron already in place. The size and shape of the window depends on both the generation number and the neuron-type. Inhibitive neurons are permitted to connect further, so their windows are larger than those associated with excitatory neurons. For a given neuron at position (i_0, j_0) the excitatory and inhibitive windows about this point are given by

$$W^E(i_0, j_0) = \{(i,j) \mid |i - i_0| \le w_g^E,\ -h_g^E \le j - j_0 \le h_g^E - |i - i_0|\},$$
$$W^I(i_0, j_0) = \{(i,j) \mid |i - i_0| \le w_g^I,\ -h_g^I + |i - i_0| \le j - j_0 \le h_g^I\},$$

where the width and height of the excitatory/inhibitive window, $w_g^{E,I}$ and $h_g^{E,I}$ depend on the generation g and are listed in the following table.

Generation g	w_g^E	h_g^E	w_g^I	h_g^I
1	10	7	10	2
2	8	9	8	7
3	8	9	6	14
4	8	9	6	21
5	8	3	6	28

Typical windows for each of the levels are depicted in Figure 3.2. Notice that the excitatory window is diamond shaped at the top and rectangular at the bottom while the inhibitive window is inverted. The excitatory window is chosen so that a neuron can connect no further than adjacent levels (and only sprout into the next level down) while the inhibitive window allows connections back to the mother neurons from any generation.

If the neuron at site (x_j, y_j) is inhibitive, it can connect to another neuron inside its window. If it is excitatory, it can connect to another neuron in its (smaller) window, or it can sprout a new neuron at an empty site inside this window. The details of these sprouting and connecting routines are described in what follows.

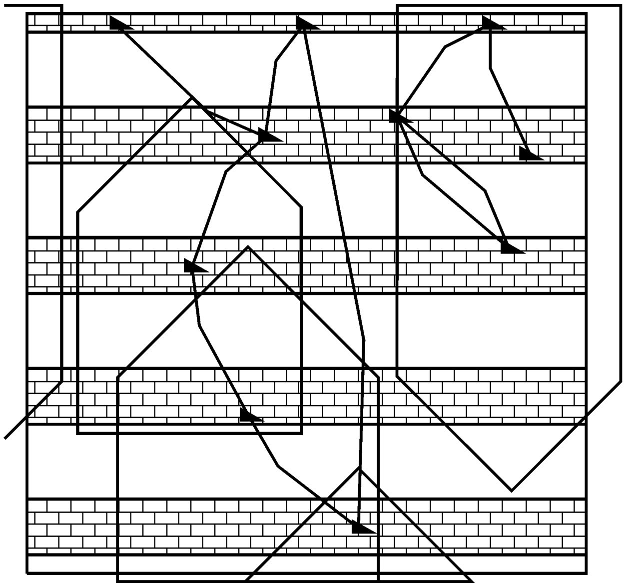

Figure 3.2. Depicted above are examples of the various excitatory and inhibitive windows or target sets. Specifically shown are excitatory target sets for levels three, four and five and an inhibitive window for level three. The characteristics of the window depend on the type and level of the neuron on which it is based.

3.3 The sprouting and connecting routines

These procedures are outlined here in detail, but they are also the content of the flow chart in Figure 3.3. For the purposes of this section, assume that we have arrived at a certain net after a number of steps, and that the activity of some of the existing neurons, either by external or internal stimulation, is above the threshold. We then loop through all the neurons and make the following tests and choices.

A. For the i-th neuron, check whether the activity is above the threshold. If no, go to the next neuron, if yes, check whether the neuron is excitatory or inhibitive. In the first case, go to B. In the second case, go to C.

B. At this point we know that the i-th neuron can either connect or sprout. Therefore define the size of the window, $W^E(x_i, y_i)$, in which to search for a suitable site. The size and shape of the window is based on the neuron type and location as described in Section 3.2. There are now three subcases.
   i) If this neuron has no axons left, $a_i(t) = 0$ then go to the next neuron.
   ii) If this neuron only has one axon left, $a_i(t) = 1$ then this neuron can only connect. A neuron cannot sprout if it only has one axon left. Go to step C.
   iii) If $a_i(t) > 1$ and the neuron is excitatory then the neuron can either sprout a new neuron or connect. Go to step D.

C. We get to this step if we are only allowed to connect which always happens for inhibitive neurons. If the i-th neuron is inhibitive, it looks at those neurons with at least one free synapse inside its target set, $W^I(x_i, y_i)$, and chooses the location that has the *maximal* potential. Alternatively, if the i-th neuron is excitatory, the target set becomes $W^E(x_i, y_i)$ and instead of maximizing the potential, it

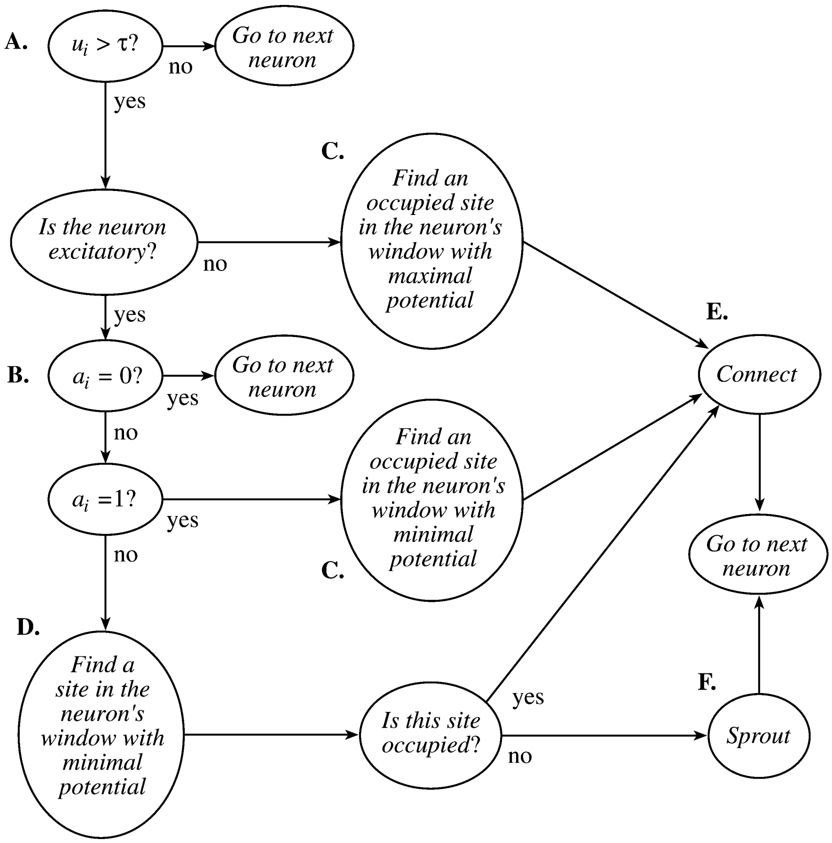

Figure 3.3. Shown here is the algorithm for the decision to sprout or connect. This is also outlined in Section 3.3.

looks for the location where the potential is *minimal*. In this search, a number of neurons are disregarded according to the following criteria:

i) We cannot already be connected to this neuron.
ii) The location to which we want to connect must have synapses left.
iii) We cannot connect to ourselves.
iv) If we are on the bottom layer we cannot connect to the bottom layer.

From the way that our potential was defined, the inhibitive neurons will typically be targeted towards neurons that already have many synaptic connections and are therefore likely to be highly active. If there are several sites which have the same maximal potential and are occupied by connection candidates, the neuron chooses one randomly for connection. Denote the suitable candidate as the j-th neuron. If no suitable neuron is found then return to step A otherwise continue to step E.

D. We get here if we can either connect or sprout. In addition, we know that the i-th neuron is excitatory. In this case we are looking for a minimum of the potential inside the target set $W^E(x_i, y_i)$. If the site under consideration is vacant we can sprout a new neuron. In this case there are two extra criteria:

i) The i-th neuron must be allowed to sprout. Recall that a neuron on level l can only sprout $G - l + 1$ times.

ii) If the position under consideration is above the current neuron's position, $y_j > y_i$, then the position is not eligible since we cannot sprout upwards.

If the site is occupied then the i-th neuron must connect. In this case we have the four criteria already listed in item C.

If there are several sites which have the same minimal potential, the neuron chooses one randomly for connection. Denote the suitable candidate as the j-th neuron. If the i-th neuron sprouts go to step F. If it connects, go to step E and if no viable site is found go back to step A.

E. Connect the i-th neuron to the j-th neuron. Section 3.4 will detail how the connection strength is determined. In the process, the number of available axonal connections for the i-th (sending) neuron is reduced by one, $a_i \to a_i - 1$, and the number of available synapses of the j-th (receiving) neuron is reduced by one, $s_j \to s_j - 1$.

F. Sprout a new neuron to position (x_j, y_j). Section 3.4 will describe how the connection weight is determined. As well, the number of free axonal connections of the i-th neuron is reduced by one, $a_i \to a_i - 1$. For the new neuron, we assign a generation number that is one less than its parent. The choice whether it be excitatory or inhibitive is made randomly, with a probability determined by the parameter ρ described in Section 3.1.

Notice that the sprouting process will automatically come to an end with the last generation, and the structure of the net will have been created by its own activity, and hence, to some degree, by the inputs which were active during the process.

3.4 The connection weights

We used a connectivity function which creates a connection weight between the i-th (sending) and j-th (receiving) neuron as

$$C(i,j) = \lambda Z_i \ \mathrm{sgm}(\mu u_i)\ \mathrm{sgm}(\mu u_j), \tag{3.2}$$

where Z_i is the neuron type of the sending neuron (-1 for inhibitive and $+1$ for excitatory neurons). In terms of the firing rates v_i (2.3),

$$C(i,j) = \lambda Z_i v_i v_j$$

and we set the connection matrix entry

$$T_{ij} = C(i,j). \tag{3.3}$$

3.5 The dynamics of the system

The dynamics of the system happen on two different time scales: A fast one for the evolution of the individual neurons' membrane potentials, given by equations (2.1), and a slow one for alterations of the net, involving the growth process, the creation of new connections and the setting of new connection strengths in accordance with (3.2) – (3.3). Once defined, connection strengths are *not* allowed to update. In practice, we solved equations (2.1) by the Matlab routine for two time units, and then stopped to update the system according to the algorithm described in Section 3.3. Any connections created at this time get the connection strength defined by (3.3). We emphasize that the already existing connectivities are not updated. Rather, they remain fixed throughout the existence of the network. It would be possible to allow synaptic plasticity via a generalized Hebb rule where already existing connectivities are updated according to

$$T_{ij}^{\text{new}} = (1-\epsilon)T_{ij}^{\text{old}} + \epsilon C(i,j)$$

with ϵ is a free parameter which determines how rapidly the connection strengths change. We did not do this as we wished to separate the effects of network growth in response to inputs from those of subsequent synaptic change. However, experiments with the learning rule included led to very similar results.

3.6 Input application

Input signals are applied directly to the lowest level of the net and felt at the upper layers through the averaging process described in Section 2. The following describes the implementation that we used. We emphasize that this is only one of many possible choices. Our first input is zero everywhere except at the grid position (5,1) (under the first mother neuron) where it has a nonzero value u_0. In our case we chose the value $u_0 = 11$. The input remains at this position for eight time units which is four of the long time cycles. This gives the network a chance to grow and alter itself in response to this input. After four long cycles the input is moved to the position (15,1) under the second mother neuron. It remains here for four cycles then moves to (25,1) completing one pass of the training cycle. This process is continued until no further growth in the net is observed. The goal of this process is to train the network to classify clearly these three input locations; in addition, the emerging networks display characteristic responses to many other inputs.

3.7 Input-output representation, graphical depiction of the net and of the potential

This is done in three windows brought up by Matlab. In one of these windows, the network window, the neurons are visible at grid-points, and their connections are displayed as piecewise straight lines (solid green for excitatory connections, dashed red for inhibitive connections). The second window, the potential window, shows the potential as a colour map. The only purpose of this map is to indicate in which direction the net will grow at a given time. The third window, the activity window, displays the activities of the mother neurons (in our case three) as a function of time; what we are looking for in this window is characteristic responses of the top layer neurons to classes of inputs.

Almost all the relevant information is contained in the network window, where the whole net and its activity is shown. In this window neurons are represented as triangles and we found it most convenient to display the connections as piecewise straight lines with one "kink" that has a rational and irrational component. This serves two purposes. The direction of the connection is from the side of the shorter segment to the longer segment. In addition, because the location of the kink has both a rational and irrational component, it is unlikely that the graphical representations of two connections will fall on top of each other.

4 Results and Discussion

We conducted extensive tests with the program and made the following observations.

4.1 The effect of ρ

In most cases, the net will grow rapidly if there is sufficient initial stimulation. The input leads to high activity of the mother neurons, to sprouting, to high activity of the next generation, which sprouts or reconnects, and so on. The process can end prematurely if too many inhibitive neurons are created early in the game. These must connect to other neurons, whose activity then gets inhibited to the point where they may not sprout new neurons anymore. High stimulation of such neurons can lead to new growth.

Typically, however, when the number of inhibitors is kept low, once a few neurons are in place, growth continues even when input is turned off. The chosen growth rule, which directs new excitatory (inhibitive) neurons to locations of lowest (highest) potential, appears to lead naturally to the formation of subnets (clusters). These clusters will usually form inhibitive connections to other clusters and back to the upper layers. The reason is the longer range of inhibitive connections. It follows that the inhibitive neurons in the lower layers are the most sensitive neurons to inputs.

We found that interesting nets, in the sense of having characteristics oscillations, were more likely to emerge if the fraction of inhibitive neurons increases from layer to layer. To this end, the parameter ρ, which is the probability that a sprouted neuron be inhibitive, was set at 0.2 for the second layer, at 0.3 and 0.4 for the next two layers, and 0.6 for the last layer which is not allowed to sprout at all, merely connect.

This description indicates that ρ is one of the most important parameters. If there is too little inhibition, such mostly excitatory nets tend to freeze in high activity states, whereas too much inhibition leads to a freeze in low activity states. Such nets still respond to input, but in a fairly trivial way-neurons directly affected by input, or directly connected to a neuron receiving input, show a response, but the rest of the net seems rather unaffected. It follows that an appropriate ratio between inhibitive and excitatory neurons is important.

4.2 The effect of the gain

The other crucial parameter appears to be the gain μ. This is not very surprising, as the gain is well-known to influence the behaviour of a neural network in sensitive ways. For low values of μ, the net tends to grow towards one with nonoscillatory stable rest states; inputs can force the net from one such steady state to another one, but it is questionable whether this is really an optimal way of information processing. First, it is not consistent with biological observations, where chaotic activity is seen in rest states ([15]) and periodic attractors seem the typical responses to inputs ([3]). Second, it may well be that chaotic rest state activity increases the information capacity of a network in ways which we do not (yet) understand.

Be that as it may, for larger values of μ, $(2 \le \mu \le 10)$ and for the choices of ρ described earlier we observed that many of our emerging nets showed characteristic *oscillatory* responses towards specific inputs; i.e., if for a grown net an input is applied at one of the 30 locations at the bottom, one, two or even all three of the mother neurons will display oscillations with characteristic amplitudes and frequencies. For most inputs, only one or two of the mother neurons will oscillate, while the others will show steady activity. This observation is satisfying inasmuch as it is consistent with biological observations;

considering oscillations as acceptable output also increases the differentiating capacity of the net dramatically, the main advantage being a vastly larger set of possible attractors, with the ability to respond characteristically to many different types of input.

4.3 What is cognitive behaviour anyway?

We have so far dodged this question, in spite of the fact that it is of central importance for our project. We admit from the outset that we have no clear definition of cognitive behaviour; we let ourselves be guided by what is seen in nature.

There are at least two different types of information-processing ability which are observed in natural neural networks. The first kind is a rather straightforward method of image-projection, in which signals are received by receptors (e.g., the retina cells in the back of the eye) and forwarded via neural connections towards a subunit of the brain, where the observed pattern is then reproduced with some accuracy. For example, this is how the visual cortex operates.

We refer to this type of information processing as trivial cognitive behaviour, for two reasons: a) it is immediate how to connect input and output neurons for the objective, namely, by high-fidelity bundles of connections which transfer an input essentially unmodulated to the output layer, and b) there is no attempt or potential for interpretation; the process is simply one of data transmission. Clearly, to produce this kind of information processing, we could have done no growing at all in our net: The three input neurons will certainly respond with different steady levels of activity to various inputs, simply by the established rule that the applied inputs are to be felt initially (with some averaging) by the mother neurons. This process will certainly define a (trivial) input-output relationship like the mentioned example of the visual system.

The second, non-trivial kind of information processing is harder to define, and we follow natural observation for a tentative definition. In biological observations, a typical reaction to a stimulus in a subunit of a neural net is a transformation from a rest state which may be noisy or chaotic ([15]) to oscillations with characteristic frequency and amplitude. It is this type of transformation which must be interpreted as a transition from a rest state to a cognitive state. Presumably, information processing via oscillatory states defines a much larger volume of all "cognitive" states than just steady states of neuron arrays ([3]), thus enhancing cognitive capacity; besides, oscillatory states may be easy to attain if certain parameter ranges and connectivity types are permitted, thus setting a natural path for evolution.

Our experiments indicate that this is a feasible path; as the results from the next section show, the mother neurons tend to respond with characteristic oscillations for most of the stimuli one can apply at the input layer; the option of a steady output is still there, but it is supplemented by a multitude of characteristic oscillations with varying frequencies and amplitudes for each neuron. The fact that this happens for an apparently open set of choices in parameter space led to the term "generic" in the title of this paper.

4.4 Results

We describe one particular realization of our network-building algorithm. Applying input to the three mother neurons in the sequence described in Section 3.6 resulted in a network that stabilized after $t = 64$ time intervals or 32 iterations with a total of 27 neurons. Snapshots of this growth process are depicted in Figure 4.1. A summary of the parameters for each of these 27 neurons at the end of this process is listed in Table 4.1.

i	(x_i, y_i)	l_i	a_i	s_i	r_i	Z_i
1	5, 30	1	1	0	4/4	1
2	15, 30	1	2	0	1/4	1
3	25, 30	1	1	0	4/4	1
4	30, 23	2	2	0	2/3	1
5	12, 23	2	1	0	-	-1
6	22, 16	3	0	0	2/2	1
7	8, 25	2	4	0	-	-1
8	4, 16	3	0	0	-	-1
9	18, 9	4	0	0	1/1	1
10	26, 25	2	1	0	0/3	1
11	29, 9	4	0	0	-	-1
12	11, 2	5	1	0	0/0	1
13	19, 23	2	0	0	-	-1
14	16, 25	2	0	0	1/3	1
15	3, 24	2	0	0	1/3	1
16	13, 16	3	0	0	-	-1
17	23, 23	2	2	0	3/3	1
18	28, 16	3	0	1	-	-1
19	17, 17	3	0	0	2/2	1
20	1, 18	3	2	0	0/2	1
21	9, 9	4	0	0	-	-1
22	24, 9	4	2	1	1/1	1
23	2, 2	5	0	2	-	-1
24	9, 18	3	2	0	2/2	1
25	14, 10	4	2	1	-	-1
26	5, 9	4	2	3	-	-1
27	21, 25	2	5	3	-	-1

Table 4.1. This table lists the parameters for all of the 27 neurons in our particular realization. Recall that a_i and s_i denote the number of unused axonal and synaptic connections for the i-th neuron respectively. The values in the r_i column are of the form p/q where p is the number of sprouts actually made out of a total possible of q. For example neuron 14 is located in level $l_i = 2$ at position (16,25). It is an excitatory neuron that has used all of its synapses, sprouted once and made four other connections since $a_{14} = 0$.

From the table one can see that this example network is fairly saturated in that there are very few synapses left for a neuron to connect to and very few axons to connect to these synapses. In addition, there are 15 excitatory and 12 inhibitive neurons. With few exceptions it is the inhibitive neurons that have a significant number of potential axonal connections remaining once the network stops growing. Recall that inhibitive neurons are not allowed to sprout.

The last image in Figure 4.1 shows the fully grown network. Two structures that seem to be essential to the development of oscillations are that all the mother neurons have at least one inhibitive connection and that there are various inhibitive neurons that have fan-like connections into the levels above them. Four of these latter structures extend from neurons 8, 16 and 18 in level 3 and neuron 11 in level 4. While similar in appearance, neuron 12 in level 5 is an excitatory neuron that has 3 inhibitive synaptic connections.

Once the network is fully grown, the time evolution of the membrane potential of the three mother neurons is observed when a delta function of strength $u_0 = 11$ is applied

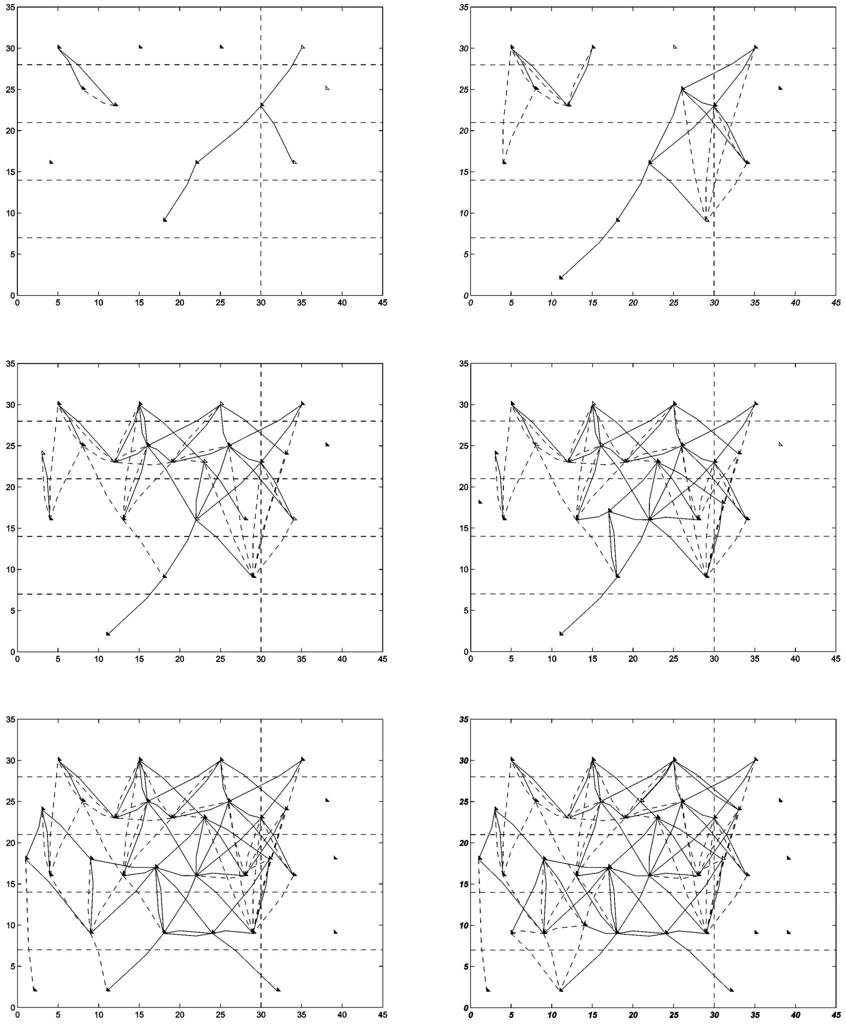

Figure 4.1. Shown here is the evolution of the network. The six figures correspond to times $t = 6$, 16, 40, 46, 56 and 64. Each iteration of the growth process takes up two time units. After $t = 64$ the network experiences no further growth.

to the bottom of the network. For a given input location the set of neurons that directly experience the input lie inside a cone as described in Section 2. Neurons outside the cone can also receive input but only through their connections. Figure 4.2 shows the membrane potentials of the three mothers with no input applied. The network tends to return to this state whenever the input application ceases.

We refer to Figure 4.3 for typical responses of the mother neurons upon stimulation at the input locations. We observed that there are large basins of attraction that extend from input locations $a = 1$ to $a = 7$ and from $a = 20$ to $a = 30$. When input is applied from $a = 8$ to $a = 19$ the network responds in a complicated fashion so that the three training locations have been effectively encoded. We emphasize that this without a Hebbian learning rule.

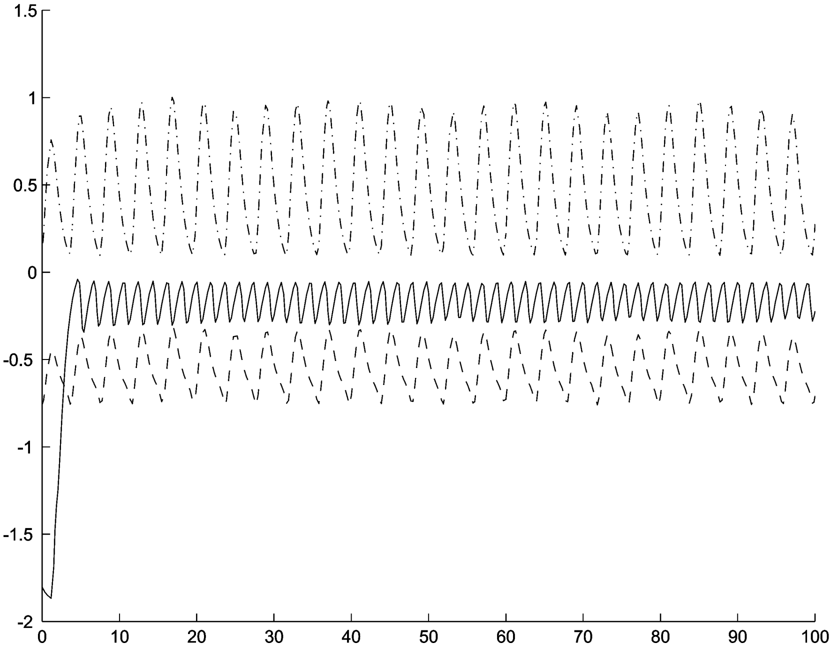

Figure 4.2. Illustrated is the response of the three mother neurons as a function of time when no input is applied to the network. The membrane potential of the neuron at position (5,30) is the solid line. The second and third neurons are at (15,30) and (25,30) and their membrane potentials are the dashed and dashed-dot lines respectively. Without any input, the activities of these two neurons oscillate in phase with each other.

By having various oscillatory responses the dimension of output space is increased. One can think of time as the extra dimension, or the frequency of each oscillation; an extra dimension is added for each output neuron. This has the clear advantage that the network grows "smarter" without adding many more neurons; the extra "intelligence" is implicitly encoded in the network structure. The main point of our work lies in the fact that this type of enhanced network performance arises generically for a range of growth rules, and it is therefore reasonable that the type of oscillatory behaviour we observe is commonplace in nature.

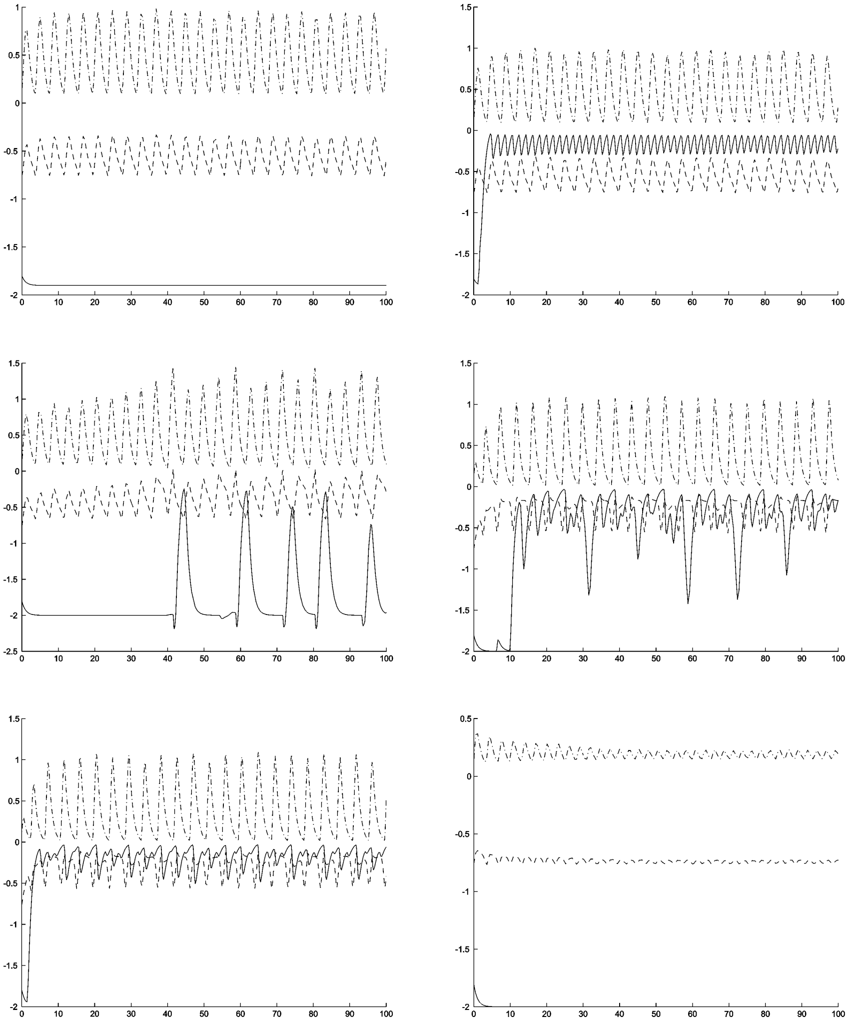

Figure 4.3. The above six figures show the response of the three mother neurons to an input of strength $u_0 = 11$ applied at locations $a = 7$, 9, 11, 13, 15 and 20. Applying input to any of the locations $a \in [1, 7]$ gives the same response as the first figure and is the same response as the rest state. As well, input applied to any $a \in [20, 30]$ results in the same response as the last figure.

5 Conclusions

We demonstrated that there are simple growth rules for a multi-layer neural network consisting of excitatory and inhibitive neurons such that the finished network will display characteristic oscillatory responses corresponding to classes of inputs. This behaviour is consistent with observed biological behaviour. We conclude that the emergence of oscillatory responses is generic in the sense that it will occur for growth rules and parameter ranges that are sufficiently general to be accessible to an evolutionary process. While our model does not directly simulate the development of biological neural systems, it demonstrates the feasibility of such a process in an idealized context and the ability to grow networks in 'sensible' ways.

References

[1] Amit, D.J. *Modeling Brain Function: The World of Attractor Neural Networks.* Cambridge University Press, Cambridge, Mass., 1989.

[2] Arbib, M.A. *The Handbook of Brain Theory and Neural Networks.* MIT Press, Cambridge, Mass., 1998.

[3] Buhmann, J.M. Oscillatory associative memories. In: *The Handbook of Brain Theory and Neural Networks.* (Eds.: M.A. Arbib). MIT Press, Cambridge, Mass., 1998, pp. 691–694.

[4] Cottet, G.H. Modèles de réaction-diffusion pour des réseaux de neurons stochastiques et déterministes. *Comptes rendus de l'Academie des sciences. Serie I, Mathematique* **312** (1991) 217–221.

[5] Edwards, R. Approximation of neural network dynamics by reaction-diffusion equations. *Mathematical methods in the applied sciences* **19** (1996) 651–677.

[6] Edwards, R. *Neural Networks and Neural Fields: Discrete and Continuous Space Neural Models.* Ph.D. Thesis, University of Victoria, Victoria, B.C., 1994.

[7] Evans, N., Illner, R. and Kwan, H. On information-processing abilities of chaotic dynamical systems. *Journal of statistical physics* **66** (1992) 549–561.

[8] Grossberg, S. Nonlinear neural networks: Principles, mechanisms and architectures. *Neural Networks* **1** (1988) 17–61.

[9] Hebb, D. *The Organization of Behavior.* Wiley, New York, 1949.

[10] Hopfield, J.J. Neurons with graded response have collective computational properties like those of two-state neurons. *Proceedings of the National Academy of Sciences* **81** (1984) 3088–3092.

[11] Hertz, J., Krogh, A. and Palmer, R. *Introduction to the Theory of Neural Computation.* Addison-Wesley, Redwood, CA, 1991.

[12] Lippmann, R.P. An introduction to computing with neural nets. *IEEE A S S P magazine* **4**(2) (1987) 4–22.

[13] Priesol, A.J., Borrett, D.S. and Kwan, H.C. Dynamics of a chaotic neural network in response to a sustained stimulus. *Technical Reports on Research in Biological and Computational Vision*, No. RBCV-TR-91-38, Dept. of Computer Science, University of Toronto, 1991.

[14] Schüz, A. Neuroanatomy in a computational perspective. In: *The Handbook of Brain Theory and Neural Networks.* (Eds.: M.A. Arbib). MIT Press, Cambridge, Mass., 1998, pp. 622–626.

[15] Skarda, C.A. and Freeman, W.J. How brains make chaos in order to make sense of the world. *Behavioral & Brain Sciences* **10** (1987) 161–195.