Design of Spatially Extended Neural Networks for Specific Applications

Rod Adams[1], Rene teBoekhorst[1], Alistair G Rust[2], Paul Kaye[1] and Maria Schilstra[1]
1. Science and Technology School,
University of Hertfordshire
Hatfield, Herts AL10 9AB, UK
Emails: {r.g.adams, r.teboekhorst, p.h.kaye,
m.j.1.schilstra}@herts.ac.uk
2. Institute for Systems Biology
1441 North 34th Street
Seattle, WA 98103, USA
Email: arust@systemsbiology.org

Abstract— **The processes and mechanisms of biological neural development provide many powerful insights for the creation of artificial neural systems. Biological neural systems are, in general, much more effective in carrying out tasks such as face recognition and motion detection than artificial neural networks. An important difference between biological and (most) artificial neurons is that biological neurons have extensive tree-shaped neurites (axons and dendrites) that are themselves capable of active signal transduction and integration. In this paper we present a model, inspired by the processes of neural development, which leads to the growth and formation of neuron-to-neuron connections. The neural architectures created have tree-shaped neurites and contain spatial information on branch and synapse positions. Furthermore, we have prototyped a simple but efficient way of simulating signal transduction along neurites using a finite state automaton (FSA). We expect that the combination of our neuronal development method with the FSA that mimics signal transfer, will provide an efficient and effective tool for exploring the relationship between neural form and network function.**

***Index Terms*—artificial neural networks, biological modelling, developmental models.**

I. INTRODUCTION

Biological neural systems are, in general, much more effective in carrying out tasks such as face recognition and motion detection than artificial neural networks (ANNs) [9]. An important difference between biological and (most) artificial neurons is that biological neurons have extensive tree-shaped neurites (axons and dendrites) that are themselves capable of active signal transduction and integration [7], [8]. It has been shown [11], [12] that axonal spike patterns generated by so-called 'bursting' neurons are strongly dependent on the shape and extension of the dendrites. Although it is unclear whether and how such spike patterns relate to neuronal function, it is widely assumed that there is such a relationship [8].

In conventional ANNs, nodes integrate instantaneous pre-synaptic input, and fire if the accumulated input exceeds a threshold. Nodes in *spiking* neural networks (SNNs), integrate input over a finite period of time, and their output depends on their current input as well as on their input history. Time is introduced through the use of delays between input and output. SNNs have been found to be particularly useful for dynamic pattern recognition of temporally encoded data [4], [23], and seem at least to be as powerful as traditional ANNs. They even require fewer neurons than conventional neural network models for the computation of certain functions [12]. Thus, the study and application of SNNs offers glimpses into the possible function of spiking and spike patterns in biological neural systems, but, as SNNs have no spatial components, they cannot provide information on the relationship between network growth, form, and function.

The performance of neural networks, whether biological or artificial, is determined by their architecture, and their potential to learn and adapt. The architecture of an ANN is largely determined by its design procedure, which is, therefore, of great importance in the selection of ANNs for specific tasks. A number of automatic design procedures exist, some constructive, building the network gradually, and some involving a pruning process [3], [14]. However, all of these procedures tend to produce relatively rigid ANNs with stereotyped architectures.

Our approach to ANN design is to simulate the development of biological neural systems, with the

aim to produce ANNs that are more dynamic and adaptive, and have a wider application range than current ANNs. In this paper we propose a scheme to integrate the computational potential of SNNs with a model of automatic ANN architecture design, in which function is related to the structure of the ANN. Therefore, we have developed a procedure for generating three-dimensional neural networks and creating inter-neuron connections that mimics embryonic neural network development [15], [17]. Furthermore, we have prototyped a simple but efficient way of simulating signal transduction along neurites.

The rest of this paper is organised as follows. Section 2 gives an overview of our developmental model, section 3 discusses the evolutionary mechanism for producing specific neural morphologies, section 4 gives some of our results and section 5 is a discussion and conclusion.

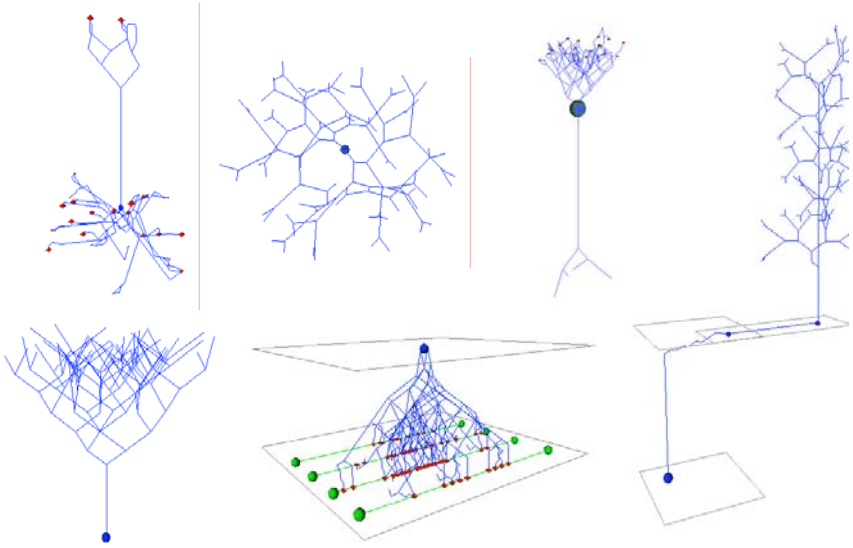

Figure1. Examples of neuron morphologies generated with the neuronal development method.

II. ANN Development Procedure

In the ANN development procedure, neurons grow within a 3-dimensional artificial 'embryonic environment', and neuron-to-neuron connectivity is created through interactive self-organisation. The embryonic environment consists of local 'chemical' gradients that are emitted by neurons. These chemical gradients affect neural outgrowth according to a small set of rules. The precise operation of these rules is controlled by a set of parameters, which are encoded into a 'genome'. Some rules specify the mechanisms that directly affect the morphology of the developing neurons (e.g. branching and pruning rules), whereas other rules specify the sequence in which genes are activated, and dictate gene-gene interaction. As a result, neurons with an identical 'genotype' can have completely different 'phenotypes', as interactions between individual neurons create differences in local chemical environment, which, in turn, lead to different patterns of growth, branching, and synapse formation.

Together these rules can be grouped into classes or phases of development:

• **Neurite Outgrowth Rules.** These rules enable neurites to extend and explore their local environments within the simulation. Rules encode how the growing tips branch in response to sensed gradient conditions. Growth is based on a simple attraction/repulsion model. The gradients of chemicals produced by dendrites attract axons and vice versa. At the same time, dendrites repel other dendrites and axons repel other axons. Synapses form when growing axons and dendrites ‘collide’.

Two principle mechanisms of neurite branching have been implemented. *Intrinsic* branches are ones that occur at genetically determined times, but where the directions taken post-branching can be mediated by

the local chemical gradients. *Interactive* splitting of a neurite is induced solely by the local environment, for example, a neurite may branch in response to chemical sources being emitted at right-angles to its current direction of growth.

• **Spontaneous Neural Activity Rules.** Once connections have formed between neurons, phases of spontaneous neural activity are used to regulate the growth rate in subsequent time steps. These particular rules are inspired by mechanisms observed during development of the mammalian retina and visual system [20].

• **Pruning Rules.** Once growth has been completed, extraneous connections and neurons are removed based on the cumulative affects of the interspersed activity phases. Overgrowth and the creation of too many initial synapses is a common phenomenon in development, representing a built in mechanism for error correction through the removal of redundant neural circuitry [1]. The model seeks to mimic such characteristics.

Whilst the developmental rules have been sub-divided into different phases, it should be stressed that the mapping from the specification of the rules (genotype) to the resulting neurons/networks (phenotype) is a complex, non-linear process due to self-organising interactions between the rules. A more detailed mathematical description of the rules can be found in [18].

A wide variety of neuron and network morphologies can be achieved by varying these parameters, some examples of which are illustrated in Figure1. The creation of individual neuron morphologies or networks does not however, have to be the result of a single phase of growth. The example in Figure 1 in the middle of the bottom row was created by growing the initial parallel fibres in the lower plane of the figure. These parallel fibres then acted as static gradient sources which attracted the descending dendrites to create the final complex structure.

III. Evolutionary Neural Development

Given the wide range of potential morphologies which the simulator can develop, the next step was to evaluate whether it was possible to direct the model to grow specific target morphologies with possible desired functionality. Essentially the capability of the model to create varieties of structure derives from the way in which the developmental rules are governed by changes in parameter values. Therefore, the task of using the developmental simulator to achieve particular architectures or functionality becomes equivalent to searching for optimal sets of developmental parameters.

The optimisation process involves searching a multi-dimensional space of all developmental parameters. The size of the search space is determined by the number of parameters under investigation and the range of values that these parameters can take. Within this space there is a hoped for region, or possibly regions, where for a given task, a set of suitable parameter values may be found. For one particular application this will be one region, whilst for another application it may be another different region. If the number of parameters is large and their potential ranges great, then the potential search space is large. The larger and more complex the search space, the longer an optimisation method will run for.

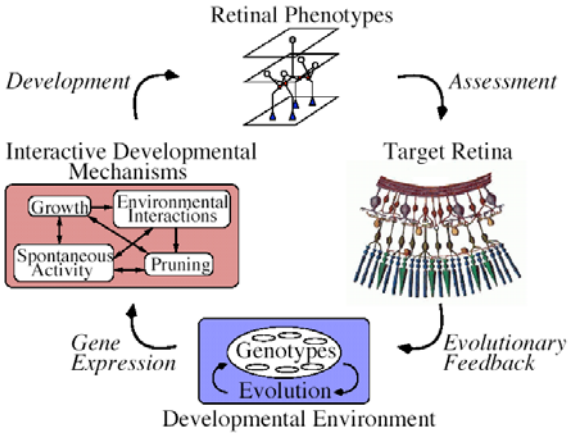

Figure2. An overview of a typical evolutionary development scenario. A population of genotypes, or sets of different developmental parameters, specifying the creation of networks or neurons, exists within an artificial developmental environment. Under the control of the developmental rules, the genotypes are decoded to form 3D neuron architectures. Each developed network, or phenotype, is then measured against a target, possibly based on a specific architecture or a desired functionality. In this example, the target architecture is a model inspired by edge-detection circuitry within the mammalian retina. This assessment of the performance of the developmental programme, provides feedback to the evolutionary process which subsequently modifies the population of genotypes. Those genotypes, or sets of developmental parameters, that produce networks with the best performances are said to be the fittest and are retained. This subset of genotypes are then bred using algorithms akin to crossover and mutation, to form a new population of prospective genotypes. The evolutionary cycle is then repeated until the evolved phenotypes hopefully meet the desired target, in this case the target retina architecture.

The genetic algorithm (GA) was chosen over other optimisation techniques (such as simulated annealing,

hill climbing and conjugate gradient) as the tool with which to search the developmental parameter space. GAs are thought to offer the best results when search spaces are large and real-valued [13]. The developmental search space can also be extremely rugged, containing many discontinuities. Other optimisation techniques are less likely to be able to traverse such spaces. The GA is incorporated into the developmental process as described and illustrated in Figure2.

An *off-the-shelf* GA implementation, GENESIS [6], was used in the evolutionary experiments. By current standards GENESIS is a simple implementation of the GA but it does include the principle evolutionary mechanisms of crossover and mutation. We specifically chose a simplistic GA to verify that it was the in-built capabilities of the developmental model which would lead to desired architectures and that it was not due to a critical reliance on the optimisation model. In essence, the GA was simply being used to fine-tune the self-organising mechanisms.

IV RESULTS

A. Evolution of Network Architectures

Our first test application was an edge-detecting retina. Two separate approaches were used to evolve a fully functional, large-scale artificial neural network capable of performing (static) edge detection [17], [18]. Since the (biological) retina has a stereotypical structure, the fitness function used by the GA rewarded correct structure in the first approach. In the second approach, the fitness function rewarded ability to detect edges on a set of sample images.

As the number of parameter values that must be evaluated in the optimization process is very large, it is necessary to evolve the genome in stages. The start of a new stage is, in many respects, comparable to the emergence of new species in the 'punctuated equilibrium' model of biological evolution [2]. 'Speciation' is achieved by the (manual) addition of new genes to a genome that is deemed close to its peak fitness. Thus, evolution starts with a genome that specifies a set of simple rules. When this genome has been adequately optimised, genes that specify more complex behaviour are added, and so on.

This modelling in stages was used in both fitness function approaches described above, as was the method of using a small-scale version of the neural network in the optimisation process to save on computational effort. The model used was a 6 by 6 grid of cone cells with correspondingly smaller test images. After evolving a small model of the retina, the resulting rules were then successfully applied to grow a retinal model capable of performing edge detection that was considerably larger (a 32 by 32 grid of cones) than the one used in the evolution [16]. Figure3 illustrates the results of the optimisation.

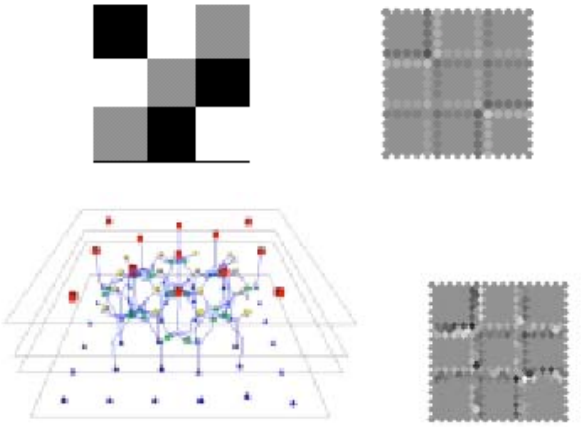

Figure3. An illustration of the neural structure grown to illustrate edge detection in a retina. This is the small retina version used in the optimisation process rather than the much larger version used as a final test. The first diagram gives the input image and the second gives the target output. The lower two diagrams illustrate the actual network produced and its output. The actual output is not an exact match for the target above as the optimisation was performed on a retina where the initial positions of the neurons are perturbed. This example demonstrates that the development process is robust to noise and adaptive to perturbations.

B. Evolution of Multi-Compartment Neurons

In the evaluation of the edge-detecting retina, all the individual neurons have dendrites (signal receivers) and a branched axon (signal transmitters), all of which have certain lengths. However, only the connectivity of the neurons was considered, not their length. In more biologically realistic models, the physical extension and signal transduction characteristics of neuronal membrane need to be taken into account as well. Using classical compartmental simulation techniques, involving multiple simultaneous ordinary differential equations (ODEs) to evaluate dendritic and axonal signal transfer, we have also been able to generate functionally realistic single bursting neurons [10]. In fact the spiking behaviour of the grown neurons was remarkably similar to that of the given stellate and pyramidal neurons, even though the artificial dendrites had shapes that were ostensibly quite different from their biological examples [16].

The process involved growing individual neurons, mapping an ODE model into the neurons, based on Mainen and Sejnowski's compartmental model [11] and then evaluating the resulting response to an induced activity pulse. In this scenario the fitness of evolved neurons was determined by analysing the frequency and spacing of generated spikes against those pulse trains measured in biological neurons.
An example of the use of the developmental simulator to evolve neurons with specific spike trains, is illustrated in Figure4.

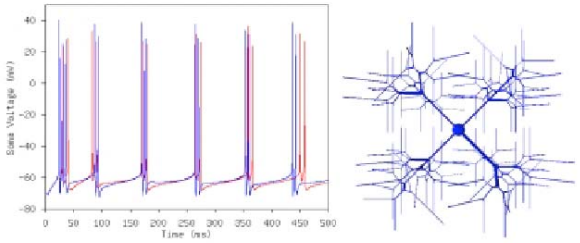

Figure4. An example of the evolution of activity spike trains in single compartmental neurons. On the left is a superposition of a required spike train (from a Pyramidal Neuron) and the spike train produced by the final evolved neuron shown on the right. The two spike trains are very similar.

V. Discussion and Conclusion

During the development of the artificial bursting neurons, we found that the computational cost of network evaluation using classical modelling techniques (all based on numerical solution of ODEs) forms a major bottleneck. Therefore, we have developed a discrete finite state automaton (FSA) model of neural signal transduction, which is, in important respects, behaviourally equivalent to the current models, and that, we hope, will permit faster evaluation of 3-dimensional neuronal networks [21].
The FSA neuron model consists of many compartments, arranged in tree structures (dendrites, axons) which are connected at their roots to a 'soma', and via synapses to neighbouring neurons (see Figure5). Passive signal transduction along the trees is modelled as a diffusional process; active signal transduction occurs in compartments that mimic the dynamic behaviour of classic two-equation models of excitable media. The FSA model can emulate the most important characteristics of both passive as well as active neurites [19].

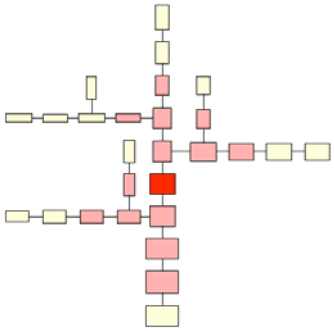

Figure5. A diagram of a FSA model of the tree structure of the compartments of a neuron. The next state of the black compartment is determined by the accumulated excitation of its neighbourhood. The shaded region illustrates a radius 3 neighbourhood.

In passive dendrites, a brief excitatory pulse administered to a single unit distributes itself over the whole structure, whereas in active neurites it generates two excitation waves ('action potentials') that move in opposite directions. Pulse propagation along a branched structure is generally secure for waves spreading towards distal regions, but tends to be blocked for waves travelling in the opposite direction.
An important aspect of active wave propagation in extended tree-like structure is that the waves interact when they coincide at a junction of two branches. Waves may annihilate or reinforce each other, dependent on their phase. Upon sustained excitation of the synapses connected over their branches, dendrites will produce complex spiking patterns that are typical for the dendrite structure, and the exact position of the synapses on the branches [5].
In a system in which branching patterns and spatial distribution of synapses affect the timing of, and interaction between signals, the morphology of a developed neural network will determine the delays (and hence arrival time) of the input signals. The smaller the difference between the arrivals, the larger the resulting post-synaptic signal and the earlier firing occurs.
By feeding the output back to the synaptic efficacy, this type of spike-timing-dependent synaptic plasticity (STDP) increases the likelihood that different synaptic inputs arrive together in a cluster [22]. Therefore, their ability to evoke post-synaptic signals is increased as well and by cooperatively generating post-synaptic signals, such a cluster can grow stronger, while weakening other synapses that are not part of the cluster. In other words, different synapses are

automatically forced to compete for control of the timing of post-synaptic signals. Besides being a desirable property in itself (it forestalls the instability inherent to Hebbian learning), this and other self-organising features will be exploited in further work to create artificial neural networks for improved dynamic pattern recognition (as prerequisites for motion detection and time series prediction).

We have shown that our neuron development method can be used to create single neurons, as well as neural networks with functional characteristics akin to biological examples. We have also demonstrated that our signal transfer FSA reproduces the key characteristics of signal transfer in biological neurons. However, the signal transfer FSA method still needs further evaluation and testing, and the development method has not yet been combined with the signal transfer FSA.

We expect that the combination of our neuronal development method with the FSA that mimics signal transfer will provide an efficient and effective tool for exploring the relationship between neural form and network function. Such a tool could be used for statistical investigations on neural spiking characteristics as a function of neuronal shape, synapse positions, and excitation patterns, as well as for rapid evolution of networks that perform specific functions. Thus, we aim to use the combined methodologies to investigate the principles that govern the functioning of biological neural networks, and also to design artificial neural networks with novel functions. In this last respect, we shall focus on networks that can deal with time-dependent processes.

REFERENCES

[1] Davies, A. M. Intrinsic programmes of growth and survival in developing vertebrate neurons. *Trends in Neurosciences*, 1994, Vol.17 No.5:195–198.

[2] Eldredge, N. and S.J. Gould, Punctuated equilibria: an alternative to phyletic gradualism, in *Models In Paleobiology*, T.J.M. Schopf, Editor. 1972, Freeman, Cooper and Co.: San Francisco, CA.

[3] Fahlman, S.E. and C. Lebiere, The cascade-correlation learning architecture, in *Advances in Neural Information Processing Systems 2*, D.S. Touretzky, Editor. 1990, Morgan Kaufman: San Mateo, CA. p. 524-532.

[4] Gerstner, W., R. Kempter, J.L.v. Hemmen, and H.A. Wagner, A neuronal learning rule for sub-millisecond temporal coding, *Nature*, 1996. 383: p. 76-78.

[5] Goodman, C.S. and C.J. Shatz, Developmental mechanisims that generate precise patterns of neuronal connectivity. *Cell*, 1993. 72: p. 77-98.

[6] Grefenstette, J. J. (1990). GENESIS Version 5 ftp://www.aic.nrl.navy.mil/pub/galist/source-code/ga-source

[7] Koch, C., Computation and the single neuron, *Nature*, 1997. 385: p. 207-210.

[8] Koch, C., *Biophysics of computation. Information processing in single neurons*. 1999, New York: Oxford University Press.

[9] Luckman, A.J., N.M. Allison, A.W. Ellis, and B.M. Flude, Familiar face recognition: A comparative study of a connectionist model and human performance, *Neurocomputing*, 1995. 7: p. 3-27.

[10] Mainen, Z.F. and T.J. Sejnowski, Reliability of spike timing in neocortical neuron,. *Science*, 1995. 268: p. 1503-1506.

[11] Mainen, Z.F. and T.J. Sejnowski, Influence of dendritic structure on firing pattern in model neocortical neurons. *Nature*, 1996. 382: p. 363-366.

[12] Maas, W., Networks of spiking neurons: the third generation of neural network models, *Neural Networks*, 1997. 10: p. 1659-1671.

[13] Mitchell, M., *An Introduction to Genetic Algorithms*, 1998, Cambridge, MA: MIT Press.

[14] Reed, R., Pruning Algorithms - a survey, *IEEE Transactions on Neural Networks*, 1993. 4(740-747).

[15] Rust, A.G., R. Adams, S. George, and H. Bolouri. Developmental Evolution of an Edge Detecting Retina, *8th Int. Conf. Artificial Neural Networks* (ICANN98). 1998. Skovde, Sweden: Springer.

[16] Rust, A.G. and R. Adams. Developmental evolution of dendritic morphology in multi-compartment neuron model, *9th International Conference on Artificial Neural Networks* (ICANN99). 1999, Edinburgh.

[17] Rust, A.G., R. Adams, S. George, and H. Bolouri, Towards Computational Neural Systems through Developmental Evolution, in *Emergent Neural Computational Architectures based on Neuroscience*, S. Wermter, J. Austin, and D. Willshaw, Editors. 2001, Springer: Berlin. p. 188-202.

[18] Rust, A.G., R. Adams, M. Schilstra, and H. Bolouri, Evolving computational neural systems using synthetic developmental mechanisms, in *On Growth Form and Computers*, S. Kumar and P. J. Bentley, Editors, 2003, Elsevier, p. 353-376.

[19] Segev, I. and M. London, *A theoretical view of passive and active dendrites*, in *Dendrites*, G. Stuart, N. Spruston, and M. Hausser, Editors. 1999, Oxford University Press: Oxford, UK.

[20] Shatz, C. J. Role for spontaneous neural activity in the patterning of connections between retina and LGN during visual system development. *International Journal of Developmental Neuroscience*, 1994, 12(6):531–546.

[21] Schilstra, M., A.G. Rust, R. Adams, and H. Bolouri, A Finite State Automation Model for Multi-Neuron Simulations, *Neurocomputing*, 2002. 44-46: p. 1141-1148.

[22] Song, S., K.D. Miller, and L.F. Abbott, Competitive Hebbian learning through spike-timing-dependent synaptic plasticity, *Nature Neuroscience*, 2000. 3: p. 919 - 925.

[23] Storck, J., F. Jäkel, and G. Deco, Learning spatio-temporal stimuli with networks of spiking neurons and dynamic synapses. *Neurocomputing*, 2001. 38-40: p. 935-943.