

# Developmental Evolution of Dendritic Morphology in a Multi-Compartmental Neuron Model

Alistair G. Rust and Rod Adams

Department of Computer Science, University of Hertfordshire, Hatfield, UK  
{a.g.rust, r.g.adams}@herts.ac.uk

## Abstract

Through the use of a multi-compartmental neuron simulation, Mainen and Sejnowski demonstrated that spike generation in neurons is a function of their dendritic structure [1]. In this paper we investigate the determination of dendritic morphology given a desired set of spike traces. A genetic algorithm is used to identify optimal parameters for a developmental model which simulates the growth of 3-dimensional dendrites. For two classes of neurons with different spiking behaviour, the developmental evolutionary process discovers ranges of viable dendritic morphologies which satisfactorily match the desired spike traces.

## 1 Compartmental Neuron Simulations

The field of computational neuroscience demonstrates that by reducing the levels of complexity found in biology to computational models, biologically defensible results can be achieved [2]. Detailed compartmental models of neurons for example, can be accurately simulated using a number of software packages [3, 4].

Using the NEURON simulation environment [3], Mainen and Sejnowski showed that dendritic structure can determine the pattern of spike trains in neurons [1]. In Mainen and Sejnowski's model a stimulus is injected into the soma of a neuron and is back-propagated through the neuron's dendritic tree [1, 5]. Due to the structure of the dendritic tree and the parameters of the compartmental model, spikes of neural activity are generated within the dendritic tree. The frequency and characteristics (single or multiple spikes) of spike generation were determined to be a function of the dendritic structure of the neurons.

If dendritic structure is a key arbiter of firing patterns in neurons, we are interested in

identifying dendritic structures given a desired spike train.

Adapting neural signalling in compartmental neuron models has previously been studied using evolutionary computation and other optimisation techniques [6, 7]. These experiments were performed on fixed neuron structures and changes in spike generation were the result of varying the parameters of the compartmental model. An alternative approach, adopted in this paper, is to keep the parameters of the compartmental model fixed and adapt dendritic morphology.

We are interested in the design of complex, artificial neural systems using a combination of biologically-inspired algorithms from neural development and evolutionary computation. The motivation for the work in this paper is to assess the feasibility of evolving dynamic neural signalling in artificial neurons, beyond simple 'integrate-and-fire' models. We aim to examine whether such functionality can be achieved simply by choosing arbitrary, artificial morphologies or whether there is a strong causal relationship between morphology and functionality. This paper thereby explores how spike trains in artificial neural systems can be generated through the adaption of dendritic structure.

## 2 Developmental Evolution

We have implemented a 3D model of biological development, in which neuron-to-neuron connectivity is created through interactive self-organisation [8]. Development occurs as a number of overlapping stages, which govern how neurons extend axons and dendrites, collectively termed neurites. Neurons grow within an artificial, embryonic environment, into which neurons and their neurites emit local chemical gradients. The growth of neurites is influenced by the local gradients and the following sets of interacting, developmental rules:

*Intrinsic rules* control the times at which neurites branch and the directions of growth

post-branching.

Initial neurites	: $2^p$
Growth rate	: $length + = \Delta length$
Branch times	: $branch@[fixed\ time]$
Branch directions	: $dirs\Theta$

Interactive rules enable neurites to branch in response to local developmental conditions and to prune unwanted connections.

Neurite navigation	: $\Delta gradients$
Local branching	: $func(local\ gradients)$
Neuron signalling	: $func(\zeta, distance)$
Synapse strength	: $\epsilon + = \Delta \epsilon$
Trophic effects	: $\Omega + = \Delta \Omega$
Neuron death	: $\Omega < threshold$
Synapse pruning	: $\epsilon < threshold$

The developmental rules are governed by parameters and by varying these parameters, a variety of neuron and network morphologies can be achieved. Examples of individual neurons are illustrated in Figure 1.

Evolving the developmental model for a specific network then becomes equivalent to the search for optimal sets of developmental parameters. Previously we have used a genetic algorithm (GA) to evolve developmental programmes which lead to the creation of an artificial, edge-detecting retina [9].

In the following experiments only the dendritic morphology of a single neuron was adapted. Parameters for only the intrinsic rules were evolved and the interactive rules were disabled.

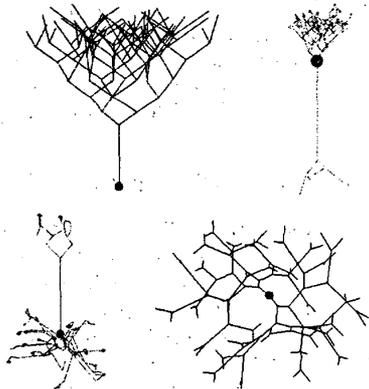


Figure 1: Examples of neurons grown using the developmental simulator. Somas are represented as spheres.

### 3 Experimental Procedure

To perform the experiments 3 software packages were integrated together; a GA, the

developmental simulator and the NEURON simulation environment. The experimental procedure is detailed below.

The GA stores a population of sets of 13 developmental parameters, where each member of the population determines how a single neuron grows dendrites. Five parameters controlled the times at which dendrites could branch, where branching is binary. These parameters specified the periods between branching and could also disable branching. A growing dendrite could thereby range from never branching up to branching a maximum of 5 times. The directions taken by dendrites post-branching were encoded in 5 related parameters. A neuron could produce 1,2,4 or 8 initial dendrites from its soma, where the initial diameter of a dendrite was also evolved (range from 1 to  $4\mu m$ ). Upon branching the diameters of new dendrites could be reduced as a proportion of the original dendrite's diameter (range from  $\frac{1}{4}$  to 1). (Soma size was taken from [1] and was not evolved.)

An individual neuron is grown within the developmental simulator and upon finishing growth at a pre-determined time period, a description of its dendritic structure is simulated within NEURON [3]. The parameters used for the multi-compartmental simulation are taken from the model described in [1]. Neurons are stimulated at their somas as previously described.

The traces produced by NEURON are then compared against an evaluation function, in this case the success with which the produced trace matches a desired spike trace. The success of simulation determines whether a neuron's developmental parameter values, stored in the genes of the GA, survive into subsequent generations. The better the performance, the greater the probability that genes will be retained.

### 4 Evaluation

The success of the developmental evolution process was evaluated against the spike generating characteristics of 2 different classes of neurons. The neurons chosen were a stellate neuron and a layer 3 pyramidal neuron taken from data in [1]. The morphologies of the neurons are shown in Figure 2.

These neurons were selected as their spike patterns were different, namely the stellate neuron produces single pulses whilst the pyramidal neuron produces bursts of multiple spikes. This presents the evolutionary process with quite different targets, allowing us to investigate whether the develop-

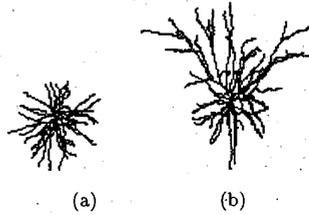


Figure 2: Neurons chosen for evaluation [1]. (a) Stellate neuron. (b) Pyramidal neuron.

mental simulator is able to generate optimal dendritic morphology for both neuron classes and whether the morphologies were obviously different.

In addition, to prevent evolved neurons from adapting to particular characteristics of a single trace, each grown neuron was evaluated against 2 traces for each class of neuron. The traces differed in the frequency with which spikes were generated and were created by simulating and recording each neuron with 2 different stimuli.

The 2 evaluation traces for the stellate neuron are illustrated in Figure 4(a) and (b). The pyramidal traces are illustrated in Figure 5. (Neurons were recorded for 500ms and 4000 data points were generated for each trace.)

The traces generated by an evolved neuron were evaluated using:

$$E = \begin{cases} A(s_a - s_t)^2 & : s_a \neq s_t \\ B \sum_{i=1}^{s_t} |e_i - t_i| & : s_a = s_t \end{cases} \quad (1)$$

where  $s_a$  is the number of actual spikes generated by an evolved neuron,  $s_t$  is the number of desired spikes in the 2 target traces,  $t_i$  is the time of the  $i$ th desired spike and  $e_i$  is the time of the  $i$ th spike of an evolved neuron.  $A$  and  $B$  are constants used to scale the error value.

In both sets of experiments  $A$  was 40 in order to strongly penalise traces containing even a single erroneous spike.  $B$  scales the error from 0 to 10 and was derived from the worst case error for incorrect times of spikes. This was calculated by determining the greatest sum of the differences between target spikes and for the 2 cases where evolved spikes all occurred at either the beginning or end of a trace.  $B$  was then set to the reciprocal of this sum, so for the stellate experiment  $B$  was  $2.67 \times 10^{-3}$  and  $1.85 \times 10^{-3}$  for the pyramidal experiment.

## 5 GA Configuration

The GA used in these experiments was GENESIS [10]. The 13 developmental parameters were encoded in the GA using 44 bits. The population size was 50 and each population was randomly initialised. The crossover rate was 0.6 using dual point crossover. The mutation rate was 0.00363, such that at each generation approximately 15% of the population would undergo a bit mutation.

The role of the GA was to minimise the error from the evaluation function.

## 6 Results

Simulations were performed on a 400MHz Pentium PC running Linux. Each population of neurons was evolved for 75 generations and then analysed. Each population of neurons took 14 hours on average to evaluate. Almost all of this time was for the individual evaluations using NEURON.

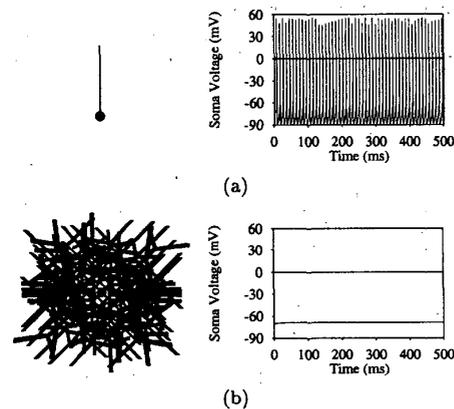


Figure 3: Examples of early generation neurons and their spike trains (evaluated upon stellate traces). (a) Single, thin dendrite:  $E = 262440$ . (b) Multiple, thick dendrites:  $E = 6760$ .

During the early generations of evolution a wide range of dendritic morphologies were explored, as illustrated in Figure 3. Figure 3(a) demonstrates how simple structure results in a high frequency spike train. Another developmental extreme is shown in Figure 3(b) where, due to the complexity of dendritic structure, no activity spikes are generated. The large associated errors result in these sets of developmental parameters being rejected for inclusion in future generations.

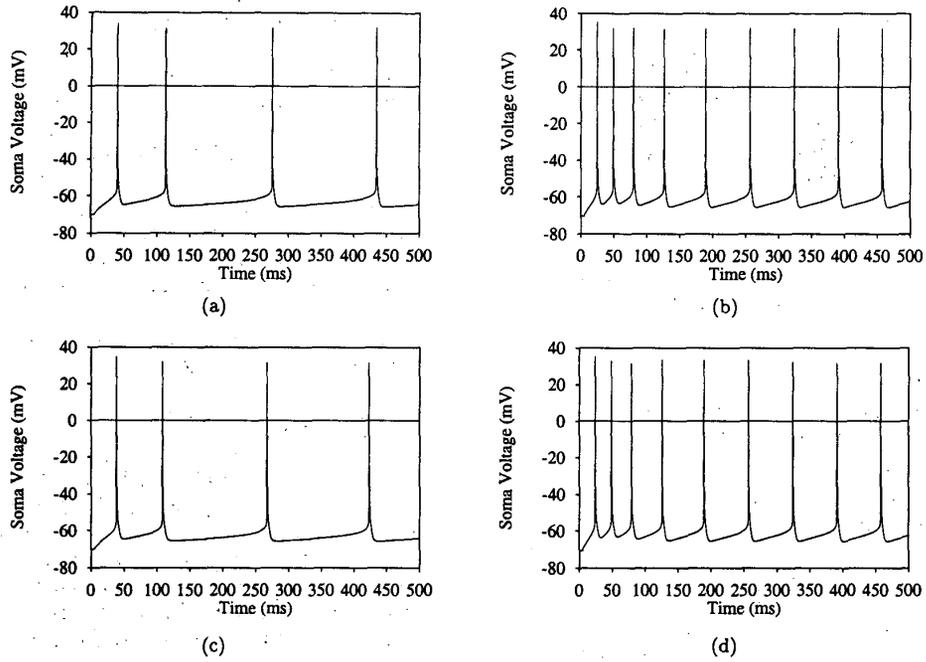


Figure 4: Spike trains for the stellate neurons. (a) and (b) are spike trains used for evaluation, whilst (c) and (d) are the spike trains produced by the best evolved neuron. Traces (a) and (c) were generated using a stimulus of 0.07. For (b) and (d) the stimulus was 0.12.

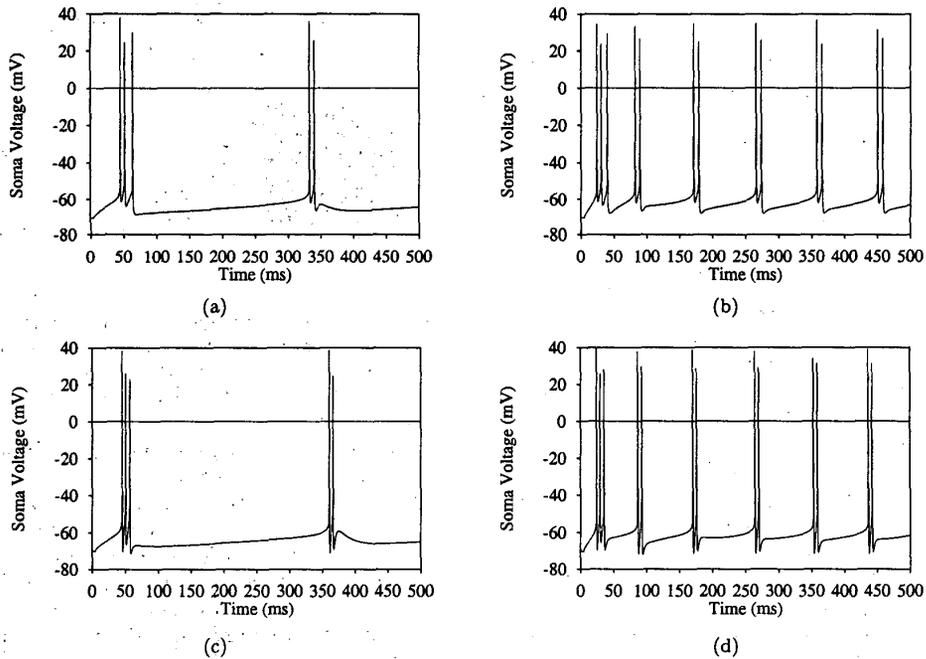
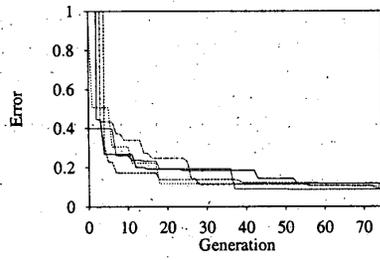
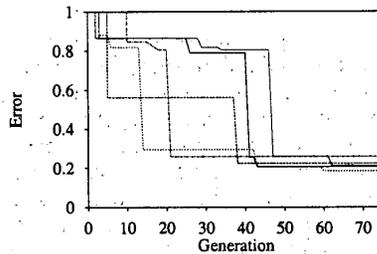


Figure 5: Spike trains for the pyramidal neurons. (a) and (b) are spike trains used for evaluation, whilst (c) and (d) are the spike trains produced by the best evolved neuron. Traces (a) and (c) were generated using a stimulus of 0.09. For (b) and (d) the stimulus was 0.16.



(a)



(b)

Figure 6: Performances of the best individuals in a population for the 2 sets of experiments. (a) records the performances of 5 experiments using the stellate spike train targets. (b) records 5 experiments using the pyramidal spike train targets.

For the stellate target traces, Figures 4(c) and (d) show the traces produced by the best evolved neuron. The best set of developmental parameters resulted in an error of 0.088 (equivalent to an average error of  $\pm 2.53\text{ms}$  per spike). All of the 5 experiments however converged to a similar error, with an average error of 0.102 ( $\pm 2.94\text{ms}/\text{spike}$ ). Figure 6(a) shows the best individual neurons in all 5 experiments.

The results for the best evolved neuron using the pyramidal neuron evaluation traces are illustrated in Figure 5(c) and (d). The best error of 0.184 ( $\pm 5.52\text{ms}/\text{spike}$ ) was higher than for the stellate experiments. Again though the experiments converged to a similar error level with an average error of 0.217 ( $\pm 6.51\text{ms}/\text{spike}$ ). The best neurons from the pyramidal neuron experiments are illustrated in Figure 6(b).

## 7 Discussion and Future Work

Figures 7 and 8 show that 2 distinct classes of dendritic structure evolved, where the morphologies of the evolved pyramidal neurons are more complex in comparison to the stellate examples. The developmental pa-

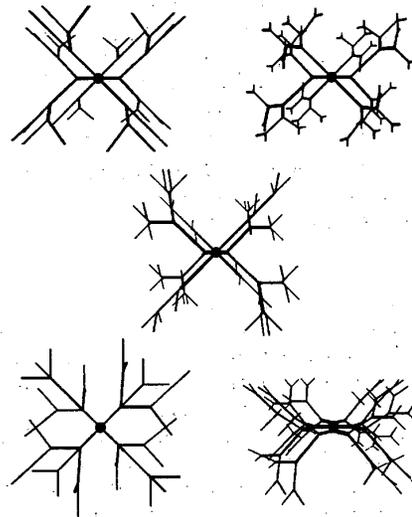


Figure 7: The morphologies of the best individual neurons from each of the 5 experiments evaluated using the stellate neuron's target traces. The central neuron was the best performing neuron.

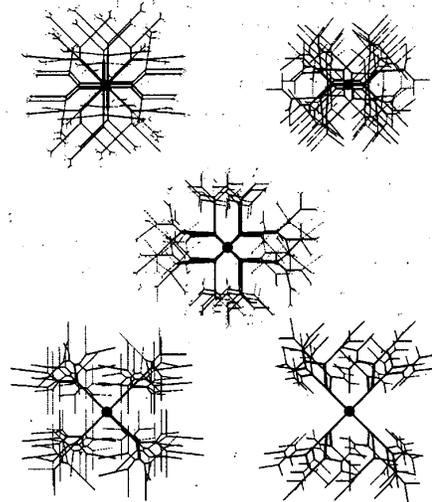


Figure 8: The morphologies of the best individual neurons from each of the 5 experiments evaluated using the pyramidal neuron's target traces. The central neuron was the best performing neuron.

Measure		Evolved Neurons	
		Stellate	Pyramidal
Total length ( $\mu\text{m}$ )	Max	4033	8963
	Min	2902	7185
	Ave	3429	8169
No. segments	Max	124	504
	Min	56	248
	Ave	97.6	352

Table 1: A summary of the differences between the dendritic morphologies of the evolved stellate and pyramidal neurons.

parameters for the pyramidal neurons instigate developmental programmes which encode a greater number of branches, resulting in longer dendritic trees containing more segments. The morphological differences between the classes are summarised in Table 1. The evolution of 2 classes of neurons shows that suitable morphologies were not achieved simply through random selection of dendritic structure.

Within the 2 classes of dendritic morphologies a range of structures can be observed. The combination of developmental mechanisms and evolution was hence able to find viable sets of parameters within the developmental search space, rather than single, unique solutions. This infers that the developmental evolution process is capable of identifying robust solutions. This is desirable if evolved neurons are to be integrated into larger networks in future work.

The difference in dendritic morphologies demonstrates that determining developmental parameters for the pyramidal neuron was a more difficult task for evolution than for the stellate neuron. This is illustrated in Figure 6 wherein the set of experiments using the stellate target traces converged in fewer generations in comparison with the pyramidal set of experiments. This is perhaps not surprising because, as can be seen in Figure 2, the morphologies of the actual biological neurons are different. Even so, by using the same initial configuration of the developmental evolution process, 2 classes of neurons were independently identified.

In future work we aim to investigate evolving dendritic structure when input stimuli are applied at the dendrites and forward propagate to the soma. In-conjunction with this, we will incorporate other features of our developmental simulator which were unused in these experiments. Namely we will add target neurons to which dendrites can grow to and explore how neuron signalling can be modified through self-adapting growth and pruning mechanisms.

We ultimately aim to incrementally build-up to simulations of multi-neuron networks.

Alternative evaluation functions need also to be sought which measure both the timing of spikes and the correlation between signals. Point-by-point correlation of traces with thousands of points, although computationally intensive, will allow us to approximate spike shapes more closely.

## 8 Conclusions

By combining a compartmental neuron simulator, a genetic algorithm and a model of neural growth, it was demonstrated how spike trains of neural activity could be adapted to match target traces by evolving the dendritic structure of neurons. For 2 different sets of neural spike trains, using the same compartmental model and the same initial configuration of the growth model, 2 distinct classes of dendritic morphologies were evolved. For each class of morphologies a range of robust solutions were identified which satisfied the evaluation criteria.

## References

- [1] Mainen ZF, Sejnowski TJ. Influence of dendritic structure on firing pattern in model neocortical neurons, *Science*, vol. 382, 363-366, 1996. NEURON demo code obtained from: [www.cnl.salk.edu/~zsch/patdemo.html](http://www.cnl.salk.edu/~zsch/patdemo.html)
- [2] Bower JM, Miller J (eds). *CNS'97*, 1997.
- [3] Hines M, Carnevale N, The NEURON simulation environment, *Neural Computation*, vol. 9(6), 1179-1209, 1997.
- [4] Bower JM, Beeman D. *The Book of GENESIS*, Springer-Verlag, 1995.
- [5] Mainen ZF, Joerges J, Huguenard JR, Sejnowski TJ. A model of spike initiation in neocortical pyramidal neurons, *Neuron*, vol. 15(6), 1427-1439, 1995.
- [6] Eichler West RM, Wilcox GL. Robust parameter selection for compartmental models of neurons using evolutionary algorithms, *Proc 5th Annual Computational Neuroscience Meeting*, 1998.
- [7] Vanier MC Bower JM, A comparative survey of automated parameter-search methods for compartmental neural models, *To appear in Journal of Comp. Neuroscience*.
- [8] Rust AG, Developmental Self-Organisation in Artificial Neural Networks. PhD thesis, Department of Computer Science, University of Hertfordshire, July 1998.
- [9] Rust AG, Adams R, George S, Bolouri H, Developmental evolution of an edge detecting retina, *Proc ICANN'98*, 561-566, 1998.
- [10] Grefenstette JJ, GENESIS 5, 1990 [ftp.aic.nrl.navy.mil./pub/galist/src](http://ftp.aic.nrl.navy.mil./pub/galist/src).