Supplemental Information for the evolution of patterns on *Conus* shells

Zhenqiang Gong[†], Nichilos J. Matzke[¶], Bard Ermentrout[‡], Dawn Song[†], Jann E. Vendetti[¶], Montgomery Slatkin[¶], George Oster[*]

[†]Department of Electrical Engineering and Computer Science, University of California, Berkeley; [¶]Department of Integrative Biology, University of California, Berkeley; [‡]Department of Mathematics, University of Pittsburgh; *Departments of Molecular & Cell Biology and ESPM, University of California, Berkeley.

Supplement A: Neural Network Model for the Conus Shell Pigmentation Patterns

1 Mathematical Formulation

In this part, we derive a shell model from a general neural model. The shell model consists of three parts: the sensory cell model, the neural model and the secretory cell model. The sensory cell model and secretory cell model are derived from first-order dynamics, and neural model is derived from the Wilson-Cowan equation (1, 2). First, we derive a continuous model. Then we discretize it to get the discrete model that is used for simulations. Finally, we discuss its relations to diffusion-reaction models.

1.1 Deriving a Shell Model from a General Neural Model

In this section we derive the model equations starting from the Wilson-Cowan model for the firing rate pattern of a general excitatory/inhibitory network (1, 2).

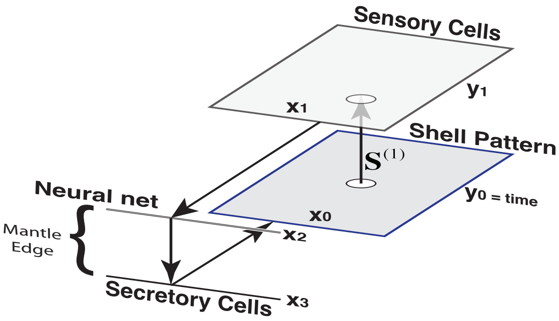


Figure S1. Schematic representation of the coordinate systems used in the derivations.

To fix our notation, we denote by 0,1,2,3 the shell pattern, sensory, neural, and secretory cells, respectively, with coordinate systems shown in Figure S1. Assume the shell is a rectangle with coordinates (x_0, y_0), where $0 \le x_0 \le L$ and $0 \le y_0 \le T_s$. $y_0 = 0$ is the growing edge. The mantle has its own coordinates (x_1, y_1), where $0 \le x_1 \le L$ *and* $0 \le y_1 \le T_M$. We assume these coordinates do not change. The sensory cells are distributed on the mantle and 'taste' the pigments. The neural cells are aligned with the growing edge along the line $y_0 = 0$ or $y_1=0$.

1.1.1 The model for sensory cells

We construct models for the sensory cells, neurons and secretory cells separately (Figure S1). The sensory cells are distributed in the mantle with coordinates (x_1, y_1). A sensory cell at (x_1, y_1) tastes the pigment on the shell at location (x_0, y_0)

In general, the activity of the sensory cells at (x_1, y_1) on the mantle satisfy the Wilson-Cowan equations:

Equation 1 $$\tau^{(1)} \frac{\partial u^{(1)}(x_1, y_1, t)}{\partial t} = -u^{(1)}(x_1, y_1, t) + S^{(1)}(\psi^{(1)}(x_1, y_1, t))$$

where $S^{(1)}(\psi^{(1)}(x_1, y_1, t))$ is the function computing the firing rate given the input:

Equation 2 $$\psi^{(1)}(x_1, y_1, t) = K^{(1)}(x_1, y_1) ** u^{(1)}(x_1, y_1, t) + M^{(1)}(x_1, y_1)$$

$K^{(1)}(x_1, y_1)$ models the recurrent connections between the sensory cells. $M^{(1)}(x_1, y_1)$ is the input to the sensory cell located at (x_1, y_1). On the neural time scale the pigment on the shell does not change, so $M^{(1)}(x_1, y_1)$ is independent of time, and is given by the double convolution:

Equation 3 $$M^{(1)}(x_1, y_1) = \int_0^L \int_0^{T_s} W^{(1)}(x_1 - x_1^{'}, y_1 - y_1^{'}) P(x_1^{'}, y_1^{'})) dx_1^{'} dy_1^{'}$$

The recurrent connections make it difficult to compute the steady state of the sensory cells, so we assume that there are no recurrent connections between sensory cells. In the steady state, we can set $\partial u^{(1)}(x_1, y_1, t) / \partial t = 0$, so that

Equation 4 $$u^{(1)}(x_1, y_1) = S^{(1)}(M^{(1)}(x_1, y_1))$$

Next, we assume that the weight kernel $W^{(1)}(x - x_1, y - y_1)$ is a two-dimensional delta function, so that a sensory cell located at (x_1, y_1) tastes only the pigment, *P*, at location $(x_0, y_0) = (x_1, y_1)$. Then we have

Equation 5 $$u^{(1)}(x_1, y_1) = S^{(1)}(P(x_0, y_0))$$

This will become the sensory input to the neural net.

1.1.2 The model for the neural net

Starting again with the steady state Wilson-Cowan equations:

Equation 6 $u_e^{(2)}(x_2) = S_e^{(2)}(\psi_e^{(2)}(x_2))$

Equation 7 $\psi_e^{(2)}(x_2) = K_{ee}^{(2)}(x_2) * u_e^{(2)}(x_2) - K_{eh}^{(2)}(x_2) * u_h^{(2)}(x_2) + M_e^{(2)}(x_2)$

Equation 8 $M_e^{(2)}(x_2) = \int_0^L \int_0^{T_M} W_e^{(2)}(x_2 - x_2', y_2 - y_2') P(x_2', y_2')) dx_2' dy_2'$

For inhibitory cells, we have similar equations. $M_e^{(2)}(x_2)$ is the sensory input to the neuron located at x_2.

Again, assume there are no recurrent connections. Then we have the steady state equation:

Equation 9 $u_e^{(2)}(x_2) = S_e^{(2)}(M_e^{(2)}(x_2))$

With a similar equation for the inhibitory neurons.

1.1.3 The model for the secretory cells

The secretory cells have first order temporal kinetics:

Equation 10 $\tau^{(3)} \dfrac{\partial u^{(3)}(x_3, t)}{\partial t} = -u^{(3)}(x_3, t) + S^{(3)}(\psi^{(3)}(x_3, t))$

Equation 11 $\psi^{(3)}(x_3, t) = K^{(3)}(x_3) * u^{(3)}(x_3, t) + M^{(3)}(x_3)$

Equation 12 $M^{(3)}(x_3) = \int_0^L (W_e^{(3)}(x_3 - x_3') u_e^{(2)}(x_3') - W_h^{(3)}(x_3 - x_3') u_h^{(2)}(x_3')) dx_3'$

The sensory inputs to the secretory cells are the weighted difference between excitatory neurons and inhibitory neurons. If there are no recurrent connections, then the steady state equation is:

Equation 13 $u^{(3)}(x_3) = S^{(3)}(M^{(3)}(x_3))$

If we assume that $W_e^{(3)}(x_3)$ and $W_h^{(3)}(x_3)$ are delta functions, then we obtain

Equation 14 $u^{(3)}(x_3) = S^{(3)}(u_e^{(2)}(x_3) - u_h^{(2)}(x_3))$

This is the pigment at the growing edge, i.e.

Equation 15 $P(x_0,0) = S^{(3)}(u_e^{(2)}(x_3) - u_h^{(2)}(x_3))$

1.1.4 The shell model

Combining the models for sensory, neural and secretory cells, we have the complete shell model:

Equation 16 Sensory cells: $u^{(1)}(x_1, y_1) = S^{(1)}(P(x_1, y_1))$

Equation 17 Neural cells: $u_{e,h}^{(2)}(x_2) = S_{e,h}^{(2)}(\int_0^L \int_0^{T_M} W_{e,h}^{(2)}(x_2 - x_2', -y_2') u^{(1)}(x_2', y_2') dx_2' dy_2')$

Equation 18 Secretory cells: $u^{(3)}(x_3) = S^{(3)}(u_e^{(2)}(x_3) - u_h^{(2)}(x_3))$

Equation 19 Pigment: $P(x_0,0) = u^{(3)}(x_0,0)$

1.2 Discrete Shell Model

The *y*-axis is time in the past. Let Δ be the spatial thickness of a single bout of pigment. We also use Δ to discretize the sensory cells. Assume $T_M = Q\Delta$, which means the mantle can sense *Q* bouts of pigments into the past. Denote $P(x_0, t)$ as the pigment at position x_0 at bout time *t* (not in real time). $A^{(1)}(x_1, s\Delta, t)$ denotes the activity of sensory cell at position $(x_1, s\Delta)$ at bout time t. $A_{e,h}^{(2)}(x_2, t)$ denotes the activity of excitatory or inhibitory neurons at position x_2 at bout time t. $A^{(3)}(x_3, t)$ denotes the activity of secretory cell at position x_3 at bout time t. Then we have

Equation 20 $P(x_0, t) = A^{(3)}(x_0, t)$

Equation 21 $A^{(3)}(x_3, t) = S^{(3)}(A_e^{(2)}(x_3, t) - A_h^{(2)}(x_3, t))$

Equation 22 $A_{e,h}^{(2)}(x_2, t) = S_{e,h}^{(2)}(\int_0^L \int_1^Q W_{e,h}^{(2)}(x_2 - x_2', -s\Delta) A^{(1)}(x_2', s\Delta, t) dx_2' ds)$

Equation 23 $A^{(1)}(x_1, s\Delta, t) = S^{(1)}(P(x_1, t - s))$

The discrete model is like:

Equation 24 $P_{n+1}(x_0) = S^{(3)}(E_n(x_0) - H_n(x_0))$

Equation 25 $E_n(x_0) = S_e^{(2)}(\sum_{j=0}^{Q-1} \int_0^L W_e^{(2)}(x_0 - x_0', j) S^{(1)}(P_{n-j}(x_0')) dx_0')$

Equation 26 $$H_n(x_0) = S_h^{(2)}(\sum_{j=0}^{Q-1} \int_0^L W_h^{(2)}(x_0 - x_0^{'}, j) S^{(1)}(P_{n-j}(x_0^{'})) dx_0^{'})$$

Figure S2 illustrates our discrete network structure. Our network is s simple feed forward network.

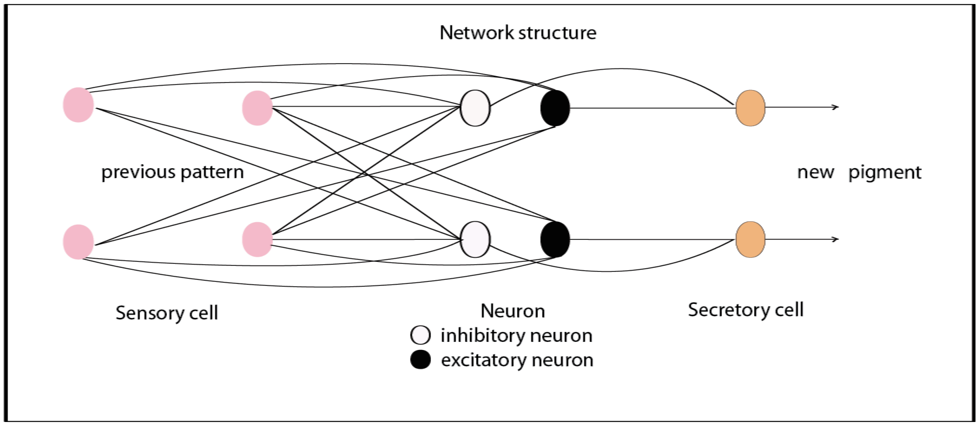


Figure S2. Illustration of network structure we use. Each neuron connects to all the sensory cells. Each secretory cell only connects to the excitatory neuron and inhibitory neuron at the same location.

1.3 Simulation Model

We assume the two dimensional space-time $W_{e,h}^{(2)}(x_2, j)$ filter is separable, i.e.

Equation 27 $$W_{e,h}^{(2)}(x_2, j) = w_{e,h}^{(2)}(x_2) v_{e,h}^{(2)}(j)$$

This widely adapted simplification is to accelerate the simulations.

In the simulation, we assume the spatial filter $w_{e,h}^{(2)}(x)$ is a Gaussian kernel

Equation 28 $$w_{e,h}^{(2)}(x) = \frac{\alpha_{e,h}^{(2)}}{2\pi\sigma_{e,h}^{(2)2}} e^{-\frac{x^2}{2\sigma_{e,h}^{(2)2}}}$$

The difference of Gaussian kernels can generate a ‘Mexican Hat’, which is necessary for pattern formation. In our model, both excitatory and inhibitory neurons have Gaussian kernels. The excitatory Gaussian kernel has a narrower variance than the inhibitory Gaussian kernel. Thus, the difference between them results in ‘Mexican Hat’. The Mexican Hat kernel generates local activation and long-range inhibition that can generate periodic patterns. Similarly, the temporal filter $v_{e,h}^{(2)}(j)$ is assumed to be a difference of exponential functions that generate a local activation and long-range inhibition in time:

Equation 29 $$v_e^{(2)}(j) = \beta_{e1}c_{e1}^j - \beta_{e2}c_{e2}^j \;\; for \;\; j \geq 0,\, \beta_{e1} = \beta_{e2} + 1,\, c_{e1}^j < c_{e2}^j$$

Equation 30 $$v_h^{(2)}(j) = \beta_{h1}c_{h1}^j - \beta_{h2}c_{h2}^j \;\; for \;\; j \geq 0,\, \beta_{h1} = \beta_{h2} + 1,\, c_{h1}^j < c_{h2}^j$$

This is equivalent to a refractory period that can generate temporal oscillations.

We assume $Q = n$, which means the mantle covers all of the previous pigment so that sensory cells can sense all the previous pigment. We define the temporal convolutions as follows:

Equation 31 $$R_{e,n}(x) = \sum_{j=0}^{n-1}(\beta_{e1}c_{e1}^j - \beta_{e2}c_{e2}^j)S^{(1)}(P_{n-j}(x))$$

Equation 32 $$R_{h,n}(x) = \sum_{j=0}^{n-1}(\beta_{h1}c_{h1}^j - \beta_{h2}c_{h2}^j)S^{(1)}(P_{n-j}(x))$$

Then we have

Equation 33 $$P_{n+1}(x) = S^{(3)}(S_e^{(2)}(w_e^{(2)}(x) * R_{e,n}(x)) - S_h^{(2)}(w_h^{(2)}(x) * R_{h,n}(x)))$$

In order to compute the temporal convolutions efficiently, we define notations:

Equation 34 $$R_{e1,n}(x) = \beta_{e1}\sum_{j=0}^{n-1}c_{e1}^j S^{(1)}(P_{n-j}(x))$$

Equation 35 $$R_{e2,n}(x) = \beta_{e2}\sum_{j=0}^{n-1}c_{e2}^j S^{(1)}(P_{n-j}(x))$$

Equation 36 $$R_{h1,n}(x) = \beta_{h1}\sum_{j=0}^{n-1}c_{h1}^j S^{(1)}(P_{n-j}(x))$$

Equation 37 $$R_{h2,n}(x) = \beta_{h2}\sum_{j=0}^{n-1}c_{h2}^j S^{(1)}(P_{n-j}(x))$$

Then we obtain the following recursive equations

Equation 38 $$R_{e,n+1}(x) = R_{e1,n+1}(x) - R_{e2,n+1}(x)$$

Equation 39 $$R_{h,n+1}(x) = R_{h1,n+1}(x) - R_{h2,n+1}(x)$$

Equation 40 $$R_{e1,n+1}(x)=\beta_{e1}S^{(1)}(P_{n+1}(x))+c_{e1}R_{e1,n}(x)$$

Equation 41 $$R_{e2,n+1}(x)=\beta_{e2}S^{(1)}(P_{n+1}(x))+c_{e2}R_{e2,n}(x)$$

Equation 42 $$R_{h1,n+1}(x)=\beta_{h1}S^{(1)}(P_{n+1}(x))+c_{h1}R_{h1,n}(x)$$

Equation 43 $$R_{h2,n+1}(x)=\beta_{h2}S^{(1)}(P_{n+1}(x))+c_{h2}R_{h2,n}(x)$$

So, Equations 33, 38, 39, 40, 41, 42 and 43 are implemented in our Matlab code to generate the patterns.

1.4 Model Parameters

The mantle's length is *L*. When we fix the number of sensory cells in our discrete model, this length only influences the interval between cells, which does not affect the patterns. So, we let *L* =1 in all our simulations.

There is a sigmoid function for the sensory cell, excitatory neuron, inhibitory neuron and secretory cell, respectively. So there are four sigmoid functions in our model. In this part, we assume all cells belonging to the same type have the same sigmoid function, i.e. all the sensory cells have the same sigmoid function, all the excitatory neurons have the same sigmoid function, etc. In later sections, we'll discuss the cases where the cells belonging to the same type have different sigmoid functions, which can generate complex patterns. The analytic form of sigmoid function is:

$$S(x)=\frac{\gamma}{1+e^{-\nu(x-\theta)}}$$

Each sigmoid function has 3 parameters, i.e. θ, ν, γ. θ is the middle point of the sigmoid function, ν is proportional to the slope at the middle point, and γ is the magnitude of the sigmoid function. We set $\gamma=1$ for all sigmoid functions in our simulations. So we have 8 free parameters for the 4 sigmoid functions.

The spatial kernels are Gaussians of the form

$$w_{e,h}^{(2)}(x)=\alpha_{e,h}^{(2)}e^{-\frac{x^2}{2\sigma_{e,h}^{(2)^2}}}$$

What's important for the pattern formation is the difference between the excitatory and inhibitory kernels, so we set the magnitude parameter $\alpha_h^{(2)} = 1$. This setting leaves us with 3 free parameters for the spatial kernels. Since there are 3 free parameters for each temporal kernel, we have 6 free parameters for temporal kernels. Thus our model is controlled by 17 free parameters, all of which have direct cellular interpretations. It turns out that the region in parameter space that generate realistic shell patterns is rather small. So the parameter search to match each shell pattern is not as difficult as the dimensionality of the parameter space might indicate.

1.5 Related Models

Cellular automata models were first used to reproduce shell patterns (3-5). Although they can generate some observed patterns, they cannot explain how these patterns arise in animal markings.

Meinhardt and his coworkers (6-10) used morphogen, or Diffusion-Reaction (DR) models to reproduce a wide variety of shell patterns. DR models are inspired by the chemical diffusion of morphogens, but there is no experimental evidence found for diffusing morphogens in pattern formation for shells. DR models can be viewed as an approximation of neural activity when only nearest neighbor neurons communicate (chapter 12.4 in (11)).

B. Ermentrout *et al* (12) and A. Boettiger *et al* (13) proposed neural models to reproduce shell patterns. Their models are somewhat different because they have different refractory terms. In (13), the refractory term is the temporal convolution of all previous pigment deposition, while refractory term in (12) is the temporal convolution of all previous pigments except the previous time period. Our model is inspired by these models, but is different in that we do not use an explicit refractory term. In the previous models, the pigment is the difference between secretory cells' activities and the refractory term. In our model, the pigment results from the net activity of the secretory cells. The previous models can only generate basic patterns, but the current model includes 'hidden' networks, and so can generate more complex shell patterns.

2 Pattern Generation

2.1 Basic Patterns

Mathematically, as stated in (13), a Turing bifurcation leads to spatial instability, which generates stripes perpendicular to the growing edge; a Hopf bifurcation generates temporal instability leading to oscillations, which generate parallel stripes. An infinite saddle-node bifurcation probably underlies the travelling waves, but we have not proven this.

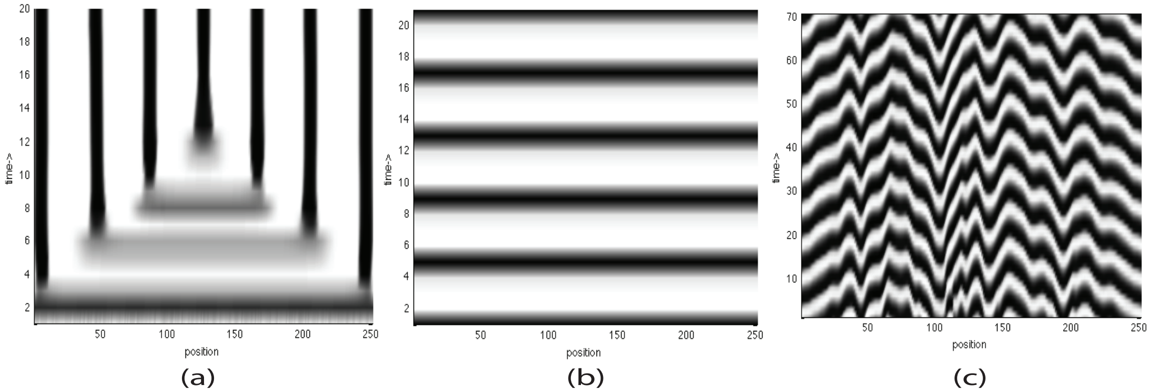


Figure S3. Bifurcations. Black indicates pigment. (a) Turing stripes. (b) Hopf bifurcation with synchronizing phase. (c) Hopf bifurcation with spatially continuously varying phases (Time increases upward)

Figure S3 (a) illustrates the formation of Turing stripes. In the very beginning, all secretory cells have deposited a very small amount of pigment. Then due to the temporal inhibition, the cells go through an unpigmented period, except for two small groups of cells on the boundaries. The two groups have reached their steady states. The activities of the two groups have lateral inhibition on their lateral regions, so their neighboring regions have no pigments. After the unpigmented period ends, an array of cells have small pigments. Near the boundary of the array, the cells have local activation on both sides, but only have one side of long-range lateral inhibition, so stripes come into being near the boundaries. *Consors* has this kind of pattern.

Hopf bifurcations can generate two categories of patterns. First, if the cells at different locations have synchronizing phase, then we get parallel lines, as shown in Figure S3 (b); second, if the cells have spatially continuously varying phase, then we get oscillations, as shown in Figure S3 (c).

If Turing bifurcation and Hopf bifurcation happen together, then we get Turing-Hopf bifurcation. There are two kinds of Turing-Hopf bifurcations depending on what the Hopf bifurcation is. If the Hopf bifurcation generates parallel lines, then we get checkerboard like patterns. And the checkerboard pattern can be in phase or out phase. In-phase checkerboard means all the checkers have the same phase in time. Out-phase checkerboard means phases of spatially neighboring checkers have 180 degrees difference. *Furvus* has in-phase checkerboard pattern. The main pattern of *tessulatus* is out-phase checkerboard. If the Hopf bifurcation generates oscillations, then we get oscillating Turing-Hopf bifurcation, checkers of which have spatially varying phases. The main patterns of *orbignyi* and *stercusmuscarum* are oscillating Turing-Hopf bifurcation.

When the secretory cell's activity represses its future activity while exciting lateral cells, we get travelling waves. When two waves collide, they may reflect, singularly annihilate or mutually annihilate. When the secretory cell has slightly wider excitatory range, then waves with changing speed emerge. *Gloriamaris*, *omaria*, *textile*, *dalli*, *episcopatus* and *aulicus* have travelling waves.

Travelling waves can also generate triangles and dots. When the speed of waves is very big at some time, then there will be a sudden stop of pigment along an array of cells, which is the base of the triangle. Then waves starting from the boundaries of the array travel back to the unpigmented region, leaving a triangular region without pigment. *Ammiralis*, *marmoreus*, and *bandanus* have triangles. Two waves starting at the same point travel outward. At some time, they change to travel inward to the unpigmented region, leaving a spot region without pigment. *Pulicarius*, *crocatus*, *arenatus* and *aurisiacus* have spots.

For Turing stripes, wider excitatory neuron spatial kernels lead to wider pigmented stripes, and wider inhibitory neuron spatial kernels generate wider unpigmented stripes. For the Hopf bifurcation, a wider excitatory neuron spatial kernel leads to wider pigmented parallel lines or oscillations, and a wider inhibitory neuron spatial kernel generates wider unpigmented parallel lines or oscillations. For waves, narrower spatial kernels can generate more dense waves. For triangles, narrower spatial kernels can generate more and smaller triangles. For dots, narrower spatial kernels can generate more and smaller dots.

2.2 Patterns with Spatial Pre-pattern

Some shells have more than one basic pattern. For example, *ammiralis* has triangles and Turing stripes. This pattern can be generated using two independent networks, one secretes pigment over another. In the current model we need use only one network to generate this complex pattern. The *ammiralis* shell is shown in Figure S4 (a). We view the triangles as the main pattern, so we find parameters to generate them. The stripes imply some parameter is different in the stripe regions. The simplest way to do this is to spatially vary the sigmoid function of the secretory cells. That is, the sigmoid function is assigned a spatial pre-pattern, and this spatial pre-pattern can be generated by a hidden network that changes the middle points of the sigmoid functions along the mantle edge. In this scenario, we say the mid-point of the sigmoid function of the secretory cell has a spatial pre-pattern. The pre-pattern and generated shell are illustrated in Figure S4 (b). In the main pattern region, the parameters generate triangles. In the stripe region, the system reaches uniform steady states. Some of the spatial pre-patterns could be generated by a third hidden networ which generates Turing stripes. In our simulations, however, we simply set the parameter's spatial pre-pattern for convenience. *ammiralis*, *tessulatus*, *laterculatus*, *aurisiacus*, *stercusmuscarum*, and *orbignyi* are generated with spatial pre-patterns.

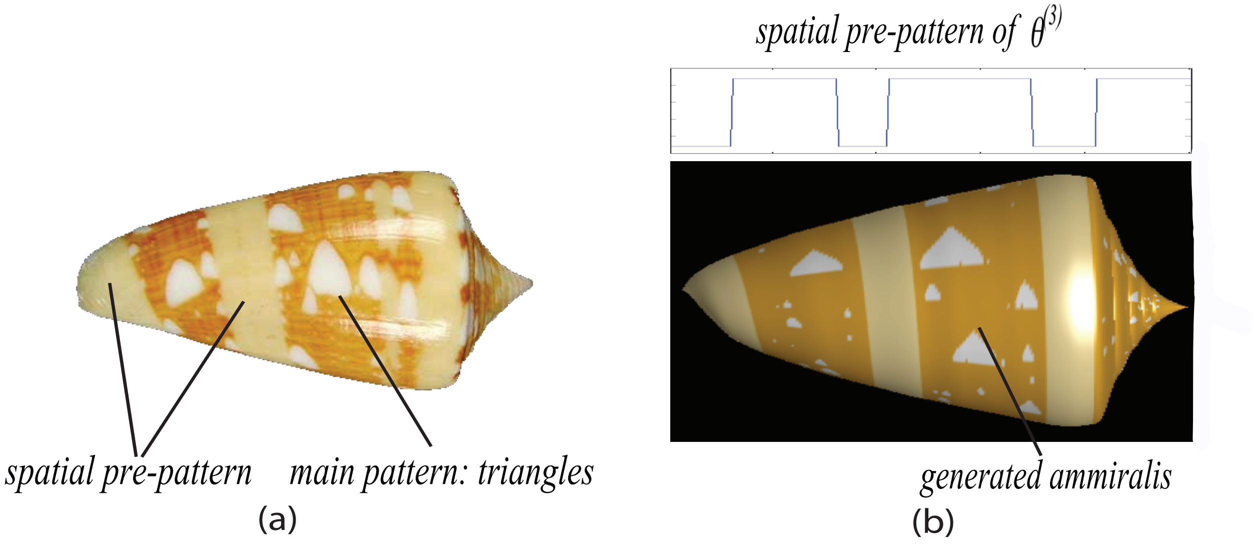


Figure S4. (a) An example showing shells with spatial pre-pattern. (b) Generated *ammiralis* and the spatial pre-pattern of $\theta^{(3)}$.

2.3 Patterns with Spatio-temporal Pre-pattern

For some more complex shells, such as *textile* shown in Figure S5 (a), a spatial pre-pattern only is not sufficient. On the *textile* shell, there are travelling waves, Hopf oscillations and Turing strips. What's more interesting is that travelling waves appear occasionally in the stripe region. One may consider using three independent networks to generate the three patterns independently, and stack them to get the *textile* pattern. However, if we do it this way, it is impossible to get travelling waves in the stripe region. Therefore, we assume that the sigmoid functions of the secretory cells have spatio-temporal pre-pattern(s). For *textile*, we view the travelling waves as the main pattern, so that there are two spatio-temporal pre-patterns: Turing stripes and Hopf oscillations. Of course, one could view the Hopf oscillations as the main pattern, and the other two as pre-patterns. Any of these different assignment of the main pattern and pre-patterns can generate this shell.

Since we assume the parameter's spatio-temporal pre-pattern is controlled by hidden network(s), we need to discuss how this is generated by the hidden network. Assume there are *N* hidden networks. Each network has its own set of sensory cells, neurons and secretory cells. We cannot see activities of the hidden networks directly. But their activities are reflected by the pre-patterns on the shell. Besides the *N* hidden networks, there is one visible network whose activity is the pattern on the shell. Another assumption we use is that each network can only sense its own activity. This assumption is rational since there are different kinds of sensory cells on our tongue, and these cells can sense different stimulus, like spicy, sweet, etc. Based on these assumptions, we propose a simple but effective method to couple these networks.

Use $P_n(x,i)$ to denote the activity of the *i*th network's secretory cell located at *x* and during the *n*th bout time. Assume $P_n(x)$ is the pattern on the shell. We make the parameter $v_n^{(3)}(x)$, slops of the middle points of the secretory cells' sigmoid functions varied by the activities of the *N* hidden networks as follows:

$$v_n^{(3)}(x) = v^{(3)} - \sum_{i=1}^{N} f_i(P_n(x,i))$$

Where $v^{(3)}$ is the basic value for $v_n^{(3)}(x)$. With this basic value only, the visible network generates the main pattern on the shell. The threshold function f_i is

$$f_i(P_n(x,i)) = \begin{cases} a_i & \text{if } P_n(x,i) \geq thres_i \\ b_i & otherwise \end{cases}$$

This threshold function means that the *i*th hidden network has only two kinds of impact on the visible network.

Figure S5 (b) shows the pre-patterns of *textile*. This pre-pattern is generated by 2 hidden networks. One generates oscillations and the other generates Turing stripes. On the main pattern region, the effect of oscillations is not strong enough to change the travelling waves pattern. In the Turing region, the effect is strong enough to change the pattern to stripes. And because of the effect of the oscillations, there are also oscillations emerging in the stripe regions. Interestingly, travelling waves emerge occasionally in the stripe regions.

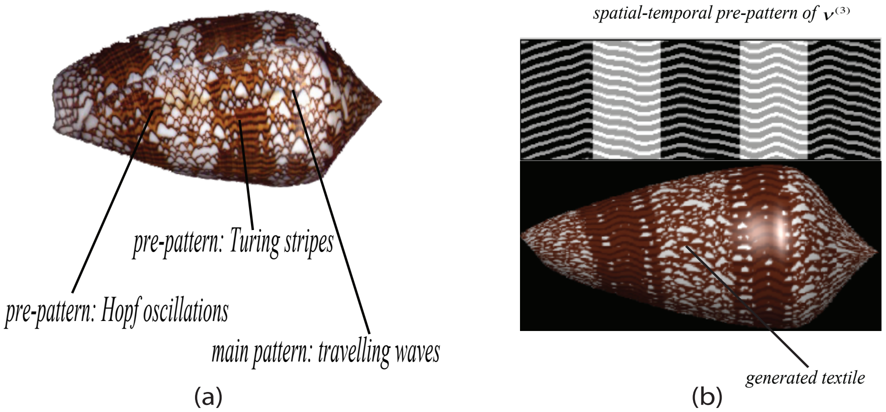
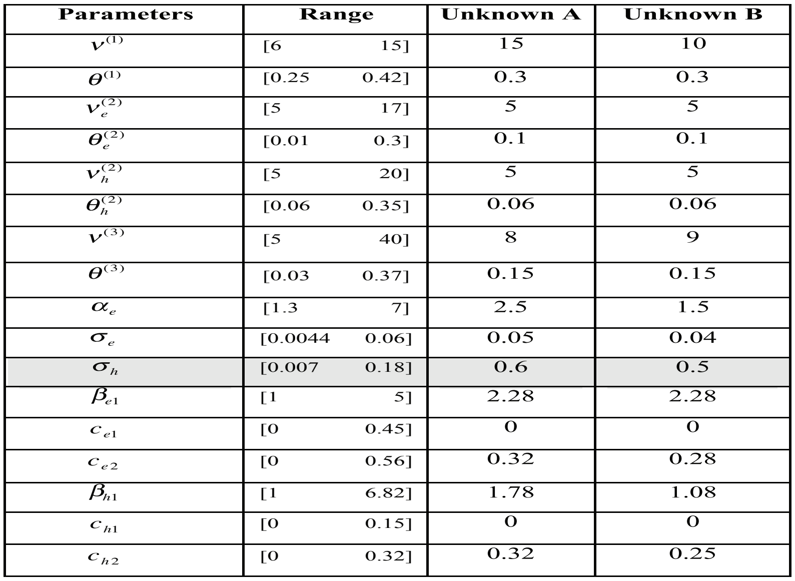

pre-pattern: Turing stripes
pre-pattern: Hopf oscillations
main pattern: travelling waves
(a)
spatial-temporal pre-pattern of $v^{(3)}$
generated textile
(b)

Figure S5. (a)An example showing the main pattern and pre-patterns of *Textile*. (b)Generated *textile* and the spatio-temporal pre-pattern of $v^{(3)}$.

3 Patterns Observed In Nature Correspond to A Small Region of Parameter Space

The ‘Mexican hat’, or ‘center-surround’ neural field is required for pattern formation. Thus the inhibition must be longer range than the activation, but with smaller amplitude than the excitation. If excitation is long range, but inhibition is short range, there will be no pattern. In addition, the strength of the excitation and inhibition must be roughly the same. Indeed, we find that real shells have excitatory and inhibitory spatial kernels of limited width, i.e. their range of excitation is fairly local, and the excitation and inhibition is roughly balanced.

Figure 3(*Right*) in the main text shows two examples of patterns generated by the neural model for which we have found no representative species. Interestingly, patterns such as these are generated when the neural net has highly ‘unbalanced’ excitation vs. inhibition. The shell patterns in Figure 3(*Right*) are unrealistic because the inhibition range is too large. We find that unrealistic patterns always have inhibition fields that are too long-range. The parameter region of realistic shells and the parameters of unknown A and B shown in Figure 3(*Right*) are illustrated in Table S1



Parameters	Range	Unknown A	Unknown B
$\nu^{(1)}$	[6 15]	15	10
$\theta^{(1)}$	[0.25 0.42]	0.3	0.3
$\nu_e^{(2)}$	[5 17]	5	5
$\theta_e^{(2)}$	[0.01 0.3]	0.1	0.1
$\nu_h^{(2)}$	[5 20]	5	5
$\theta_h^{(2)}$	[0.06 0.35]	0.06	0.06
$\nu^{(3)}$	[5 40]	8	9
$\theta^{(3)}$	[0.03 0.37]	0.15	0.15
α_e	[1.3 7]	2.5	1.5
σ_e	[0.0044 0.06]	0.05	0.04
σ_h	[0.007 0.18]	0.6	0.5
β_{e1}	[1 5]	2.28	2.28
c_{e1}	[0 0.45]	0	0
c_{e2}	[0 0.56]	0.32	0.28
β_{h1}	[1 6.82]	1.78	1.08
c_{h1}	[0 0.15]	0	0
c_{h2}	[0 0.32]	0.32	0.25

Table S1 Parameters regions and the parameters for unknown pattern A and B shown in Figure 3(*Right*)**.** The shaded row is the parameter that extends the inhibition to unrealistic size.

Supplement B: Statistical Methods and Phylogenetic Analyses

4 Software and phylogenetic data

The statistical and phylogenetic analysis was conducted using the R statistical language (14, 15) enhanced with the R phylogenetics packages APE (16), geiger (17), and phangorn (18) and custom scripts by N.J.M. (available upon request).

The phylogeny used for the analysis was that of Nam *et al*. (19). It is a well-supported phylogeny of the *Conus* species under study. The phylogeny, shown in Figure S6, is based on mitochondrial COI and rDNA sequences and on ITS2 sequences from nuclear ribosomal DNA. Nam *et al*. showed that ITS2 sequences resolved parts of the phylogeny that could not be resolved using only the mtDNA. The bootstrap values are sufficiently high that we will assume the phylogeny is correct.

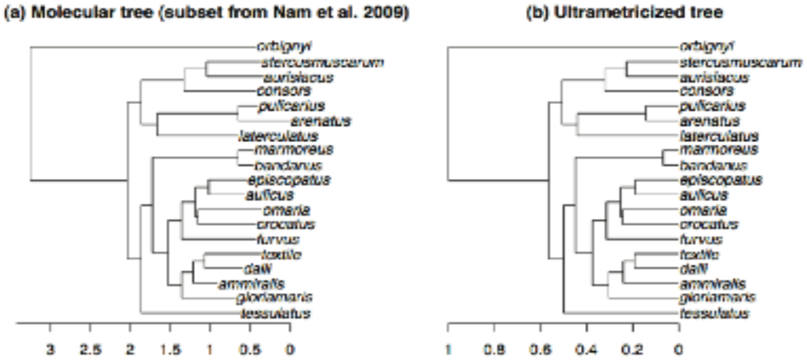


Figure S6. (a) DNA-based tree used in this study, digitized from Nam et al. (2009). Taxa not used in this study have been excluded. (b) Ultrametric tree used in the study, calculated using NPRS. Absolute time information was not need for this study, so branch lengths are in units of relative time, with the root set to age 1.

The phylogeny was digitized to Newick format using GraphClick 3.0 (http://www.arizona-software.ch/graphclick/) and an in-house R script, TreeRogue 0.1 (available at: https://stat.ethz.ch/pipermail/r-sig-phylo/2010-October/000816.html). Correspondence of the topology and branch lengths of the digitized tree to the original was verified before use. Taxa from Nam *et al*. that were not used in this analysis (*C. radiatus*, *C. parius*, *C. japideus*, *C. vimineus*, and outgroups) were dropped from the tree.

The time-calibrated ultrametric phylogeny was calculated using non-parametric rate scaling (NPRS) with the program r8s (20). Branch lengths are proportional to the total height of the tree; only branch lengths giving a relative measure of degree of shared ancestry were necessary for this study, rather than a tree calibrated to absolute time.

5 Test for phylogenetic signal in the parameter estimates

The estimation of neural network parameters is done manually through successive approximation. Before examining the matter, there was no guarantee *a priori* that there would one unique parameter solution to produce a specific shell pattern, or that the user will find, in a 19-dimensional parameter space, the single best match to the observed pattern in the living species. Additionally, it was conceivable that there might no phylogenetic signal in shell patterns of the living species either for biological reasons (rapid evolution to an equilibrium distribution of shell patterns) or technical ones (e.g., nonidentifiability of parameters of the neural network model).

To assess these assumptions, we tested whether or we could reject the null hypothesis of no phylogenetic signal in the 19 continuous parameters. A neighbor-joining phylogeny was constructed from the parameters, as follows**.** (1) each of the neural network parameters was normalized to a 0-1 scale; (2) the normalized parameters were used to calculate the pairwise Euclidean distance between each pair of species; (3) a phylogeny was inferred from the resulting distance matrix via neighboring-joining (16). The parameter-based tree was then compared to the DNA-based tree using tree-to-tree distance metrics. Two metrics were used. A measure of topological distance (considering just tree topology, and ignoring branch length) was provided by Robinson-Foulds topological distance, also known as symmetric difference (dist.topo, PH85 option in APE). This measures the number of partitions found in each tree which are not found in the other (21-23). A measure of distance that takes branch lengths into account is provided by Robinson-Foulds branch-length difference, which is the sum of changes in branch length that would have to be made to made two trees identical (24); dist.topo in APE, BHV01 option).

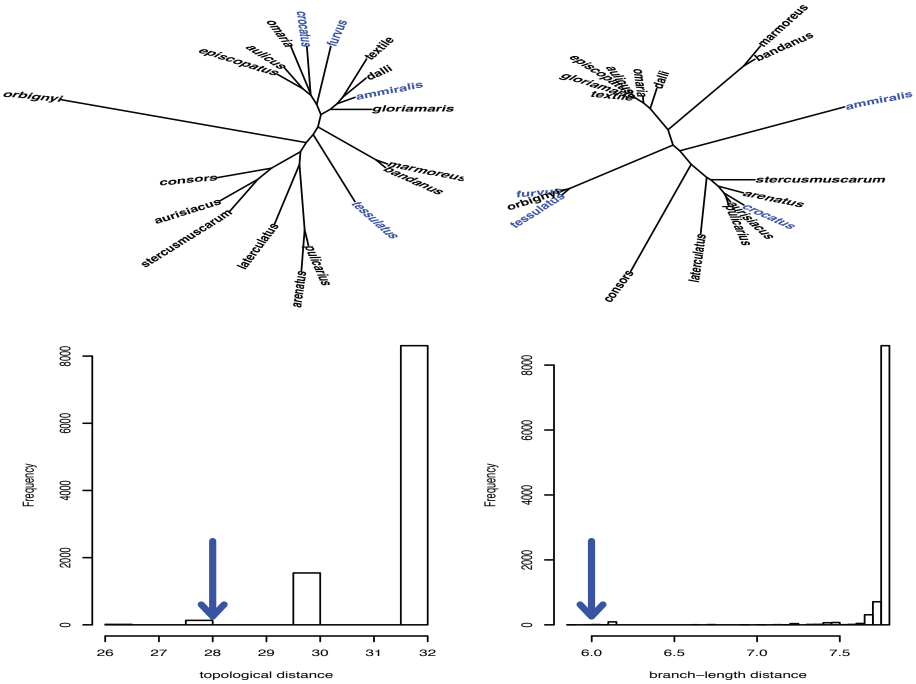


Figure S7. Top: Comparison of the DNA-based phylogeny of cone snails (left, after Nam et al. 2009, unrooted for display) and the parameter-based tree (right, this study). Species labeled in blue exhibit major changes in topological position in the parameter-based tree. Bottom: Although by eye the trees seem rather different, their statistical similarity is much greater than expected by chance. The observed tree-to-tree distances (blue arrows) are significantly smaller than expected under a null hypothesis of random similarity. Histograms show the distribution of distances between the DNA−based tree and the parameter−based tree (both ultrametricized and scaled to the same total length) under 10,000 realizations of the null hypothesis of no similarity between the trees (species names randomly shuffled in the parameter tree). Non−parametric p−value for topological distance = 0.0146; for the branch−length difference, p = 4e−04.

The null distribution of the of DNA-tree-to-parameter-tree distances was constructed by randomly shuffling the tip labels on the parameter tree and measuring the distance between it and the DNA-based tree. This was repeated 10,000 times. For each distance metric, the one-tailed p-value was obtained by comparing the rank of the observed distance between the parameter tree and the DNA tree to the 10,000 ranked distances from the null distribution. For branch-length difference, it is possible that the non-ultrametric nature of the DNA tree and the parameter tree, and the different total lengths of the trees, could bias the results, so the test (including generation of the null trees) was repeated on ultrametricized versions of the DNA tree calculated with nonparametric rate smoothing (NPRS) in r8s (20) and the parameter tree (which was midpoint-rooted, and then rescaled with NPRS), with no substantial differences in results.

The observed tree-to-tree distances, and the null distributions on tree-to-tree distances, are shown in Figure S7 for the case where both compared trees were ultrametric. For both distance metrics, the observed distance between the DNA tree and the parameter-based tree was significantly closer than expected under the null of random similarity between the trees.

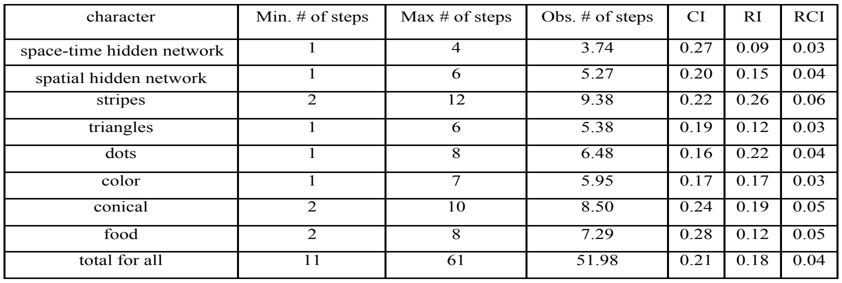
6 Phylogenetic signal in discrete characters

In order to compare our results to those that might be obtained using more traditional cladistic methods where the shell patterns are described with discrete character states, several prominent discrete shell-pattern characters were scored in the traditional cladistic manner based on photographs of the species. The characters were scored as follows: stripes: 0=absent, 1=weak, 2=strong; triangles: 0=present, 1=absent; dots: 0=absent, 1=present; color: 0=black and white only, 1=brown/orange and white. One shell shape character was also scored: conical shape: 0=strongly rounded, 1=weakly rounded, 2=no rounding (triangular cone). Additionally, prey preference was scored according to Nam et al.'s descriptions (0=piscivore, 1=verminivore, 2=molluscivore, ?=unknown). These characters were mapped onto the phylogeny using parsimony and maximum likelihood, which gave similar results. The congruence of these characters with the phylogeny was measured with consistency index (CI), retention index (RI), and rescaled consistency index (RCI), and the significance of these results was assessed by comparison to null distributions of these statistics generated by 1000 reshufflings of the tip data.

Phylogenetic signal in discrete characters. Parsimony-based summary statistics for the 8 discrete characters are shown in Table S2-S5.

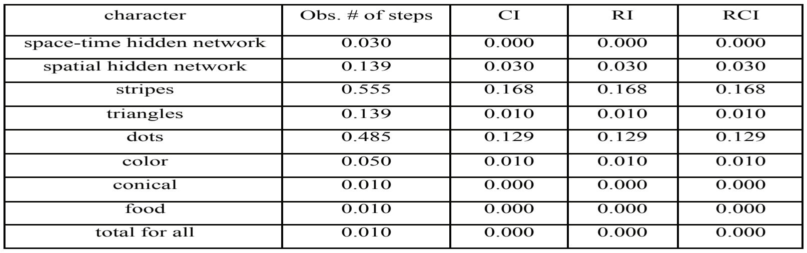
character	Min. # of steps	Max # of steps	Obs. # of steps	CI	RI	RCI
space-time hidden network	1	4	2	0.50	0.67	0.33
spatial hidden network	1	6	4	0.25	0.40	0.10
stripes	2	12	9	0.22	0.30	0.07
triangles	1	6	4	0.25	0.40	0.10
dots	1	8	6	0.17	0.29	0.05
color	1	7	4	0.25	0.50	0.13
conical	2	10	5	0.40	0.63	0.25
food	2	8	2	1.00	1.00	1.00
total for all	11	61	36	0.31	0.50	0.15

Table S2 Parsimony summary statistics (CI, RI, RCI) for discrete characters along with the inputs to these statistics. Values closer to 1 indicate more congruence between the character and the phylogeny. However, these must be compared to null distributions to determine if the observed values of the statistics are higher than would be expected under the null hypothesis of no phylogenetic signal (see Table S3 and Table S4).



character	Min. # of steps	Max # of steps	Obs. # of steps	CI	RI	RCI
space-time hidden network	1	4	3.74	0.27	0.09	0.03
spatial hidden network	1	6	5.27	0.20	0.15	0.04
stripes	2	12	9.38	0.22	0.26	0.06
triangles	1	6	5.38	0.19	0.12	0.03
dots	1	8	6.48	0.16	0.22	0.04
color	1	7	5.95	0.17	0.17	0.03
conical	2	10	8.50	0.24	0.19	0.05
food	2	8	7.29	0.28	0.12	0.05
total for all	11	61	51.98	0.21	0.18	0.04

Table S3 Means of summary statistics calculated on 1000 draws from the null hypothesis where character states have been randomly shuffled on the tips (no phylogenetic signal).



character	Obs. # of steps	CI	RI	RCI
space-time hidden network	0.030	0.000	0.000	0.000
spatial hidden network	0.139	0.030	0.030	0.030
stripes	0.555	0.168	0.168	0.168
triangles	0.139	0.010	0.010	0.010
dots	0.485	0.129	0.129	0.129
color	0.050	0.010	0.010	0.010
conical	0.010	0.000	0.000	0.000
food	0.010	0.000	0.000	0.000
total for all	0.010	0.000	0.000	0.000

Table S4 Non-parametric p-values (one-tailed) of summary statistics, based on the rank of the observed statistic amongst the ranks of the 1000 null draws of the statistic summarized in Table S3. The null hypothesis of only random congruence with the phylogeny cannot be rejected at the p=0.05 level for the stripes and dots characters. No Bonferroni correction for multiple tests has been applied; if it is, then s3n_spd, triangles, and color also fail to reject the null at the p<0.05 significance level.

Two discrete characters, stripes and dots, fail to reject the null hypothesis of only random congruence with the phylogeny, with one-tailed non-parametric p-values > 0.05 (Table S4). For each statistic, a value of 1 indicates perfect congruence with the phylogeny, 0 means no congruence. The statistics for prey preference are all 1, indicating perfect congruence with phylogeny (Table S4). And the discrete characters, all taken together, also exhibit significant congruence (Table S4). However, the variability in stripes and dots indicates substantial homoplasy in these characters, despite their obviousness to human observers. This is likely an indication that multiple convergent pathways able to produce these patterns.

7 Test of character independence

Before estimation of ancestral states was attempted, the parameter estimates for living species were examined for correlation structure, as the simplest methods of ancestral character estimation assume that each character is independent. Using standard correlation analysis (Pearson's product-moment correlation, p-values produced with t-test), only 10 of 171 parameter pairs exhibited statistically significant correlation ($p < 0.05$ after Bonferroni correction for multiple tests).

Standard correlation analysis is dubious when the data may have correlation due to shared phylogenetic structure, so the correlation analysis on the 19 species was repeated on the 18 available phylogenetically-independent contrasts (PICs) (25). PICs were calculated using an ultrametricized phylogeny derived from the Nam *et al.* molecular tree by nonparametric rate smoothing (NPRS) (20). Using PICs, only 7 of 171 parameter pairs were statistically significantly correlated. Of these, two pairs were the β parameters of the temporal kernels of the excitatory and inhibitory neurons (β_{e1} and β_{e2} formed one pair, and β_{i1} and β_{i2} the other), which were always perfectly correlated in the estimates made on living species (the first parameter of the pair is always 1 unit higher than the second). The others were correlations of about 0.75-0.9 between parameters of the response functions of the excitatory and inhibitory neurons and the parameters of the temporal kernels for the excitatory and inhibitory neurons. A more elaborate analysis might take these correlations into account, or even take into account weak correlation structure that exists despite statistical insignificance, but for the present purpose of assessing the feasibility of integrating the developmental model with phylogenetic methods, statistical simplicity was preferred, and was judged to be a reasonable approximation given the overall weak correlation structure.

8 Model Selection

Tests for trait correlations using Phylogenetic Independent Constrasts (PIC), and the estimation of ancestral states, are most easily performed if the trait data can be modeled as evolving under a Brownian motion process. Under Brownian motion, traits wander without limit such that the expected variance (σ^2) between lineages increases as a linear function of phylogenetic distance.

This may be a valid approximation within a relatively closely-related clade where the traits have not yet run up against intrinsic limits (26-28). The Brownian motion model was tested against a variety of other models of continuous trait evolution using parametric methods (likelihood ratio test (LRT) and Akaike Information Criterion (AIC), (29) as well as nonparametric methods; the latter are expected to be more robust in the situation where the low number of taxa (here, 19) mean that the asymptotic assumptions of maximum likelihood inference may not be met in full. (30)

Seven models for the evolution of continuous traits were compared. The likelihood of the trait data under of each of the models was calculated using the R package geiger. (17) The Brownian motion model has two parameters (global mean and rate of variance increase); all of the others, except white noise, add one additional parameter so as to model stabilizing selection (Ornstein-Uhlenbeck, OU), lack of phylogenetic signal on internal branches (lambda), speciational or punctuated evolution (kappa), increases or declines in rate across the tree (delta), or an early burst of evolutionary change which then declines. White noise assumes no phylogenetic signal and simply models the data as a normal distribution with a mean and variance. (17, 27, 29, 30) The log-likelihoods for the neural network parameter data under each model are given in Table S5.

Model	s1n	s1t	s2en	s2et	s2in	s2it	s3n	s3t	sf2ea	sf2ed	sf2id	tf2eb1	tf2ec1	tf2eb2	tf2ec2	tf2ib1	tf2ic1	tf2ib2	tf2ic2	kz	kr	turn	N	T
Brownian	-46.9	28.6	-50.3	24.6	-54.7	11.5	-68.1	14.1	-30.1	55.5	29.1	-23.8	13.4	-23.8	10.1	-35.2	30.4	-35.2	17.5	61.0	40.8	-44.7	-100.4	-136.1
O-U	-46.9	28.6	-50.3	24.8	-54.3	13.3	-68.1	14.3	-30.1	56.8	31.4	-23.8	13.4	-23.8	10.1	-31.4	30.4	-31.4	17.9	61.5	43.5	-42.2	-100.4	-134.7
lambda	-46.9	28.6	-50.3	24.6	-54.7	12.2	-68.1	14.1	-30.1	56.7	30.0	-23.8	13.4	-23.8	10.1	-31.5	30.4	-31.5	17.5	61.0	43.1	-42.3	-100.4	-134.7
kappa	-46.9	28.6	-49.1	24.6	-54.7	11.5	-68.1	14.1	-29.7	55.5	29.5	-23.8	13.4	-23.8	10.1	-35.1	30.4	-35.1	17.5	61.0	41.4	-44.5	-100.4	-136.1
delta	-46.8	28.7	-50.1	25.0	-54.1	12.9	-91.2	14.4	-29.9	56.7	30.7	-23.8	13.7	-23.8	10.2	-33.2	30.5	-33.2	18.1	61.7	42.5	-43.1	-1444.7	-51452.1
early burt	-46.9	28.6	-50.3	24.6	-54.7	11.5	-68.1	14.1	-30.1	55.5	29.1	-23.5	13.4	-23.5	10.1	-35.2	30.5	-35.2	17.5	61.0	40.8	-44.7	-99.9	-136.1
white noise	-48.8	27.0	-54.4	23.5	-57.0	12.2	-71.9	13.9	-33.8	56.7	30.0	-27.8	9.4	-27.8	7.9	-31.5	28.1	-31.5	17.2	58.8	43.1	-42.3	-105.9	-134.7

Table S5 Log-likelihoods of the observed shell pattern parameters as explained under different models. Parameter abbreviations on the x−axis correspond to those in Table S9, with underscores removed and with the last character being the Latin equivalent of the relevant Greek symbol.

Likelihood-Ratio Tests (LRT). All of the models except for white noise contain Brownian motion as a special case and thus can be compared to Brownian motion via a LRT, which is chi-squared distributed with 1 degree of freedom. (31) The LRT fails to reject the Brownian motion model at the 0.05 significance level for 16/19 neural network parameters. When Brownian motion loses, it loses to the O-U model, which suggests that stabilizing selection is removing phylogenetic signal (Table S6).

Model	s1n	s1t	s2en	s2et	s2in	s2it	s3n	s3t	sf2ea	sf2ed	sf2id	tf2eb1	tf2ec1	tf2eb2	tf2ec2	tf2ib1	tf2ic1	tf2ib2	tf2ic2	kz	kr	turn	N	T
O-U	1.000	0.900	0.741	0.515	0.335	0.055	1.000	0.556	0.864	0.107	**0.031**	1.000	0.771	1.000	0.938	**0.006**	1.000	**0.006**	0.331	0.303	**0.018**	**0.027**	0.999	0.090
lambda	1.000	1.000	1.000	1.000	1.000	0.232	1.000	1.000	1.000	0.134	0.177	1.000	1.000	1.000	1.000	**0.006**	1.000	**0.006**	1.000	1.000	**0.029**	**0.028**	1.000	0.090
kappa	1.000	1.000	0.118	1.000	1.000	0.837	1.000	1.000	0.397	1.000	0.360	1.000	1.000	1.000	1.000	0.747	1.000	0.747	1.000	1.000	0.243	0.513	1.000	1.000
delta	0.670	0.684	0.475	0.357	0.246	0.089	1.000	0.485	0.611	0.129	0.071	0.815	0.470	0.815	0.629	**0.045**	0.913	**0.045**	0.252	0.215	0.065	**0.076**	1.000	1.000
early burst	0.884	1.000	1.000	1.000	1.000	1.000	0.787	1.000	1.000	1.000	1.000	0.395	1.000	0.395	1.000	1.000	0.878	1.000	1.000	1.000	1.000	1.000	0.324	1.000

Table S6. P-values of likelihood-ratio tests (chi-squared, 1 d.f.) for each model compared to the Brownian motion model. Tests that are significant at the 0.05 level are bolded. No correction for multiple testing was made here; a Bonferroni correction for 95 tests makes all results non-significant at the 0.05 level. Parameter abbreviations on the x−axis correspond to those in Table S9, with underscores removed and with the last character being the Latin equivalent of the relevant Greek symbol.

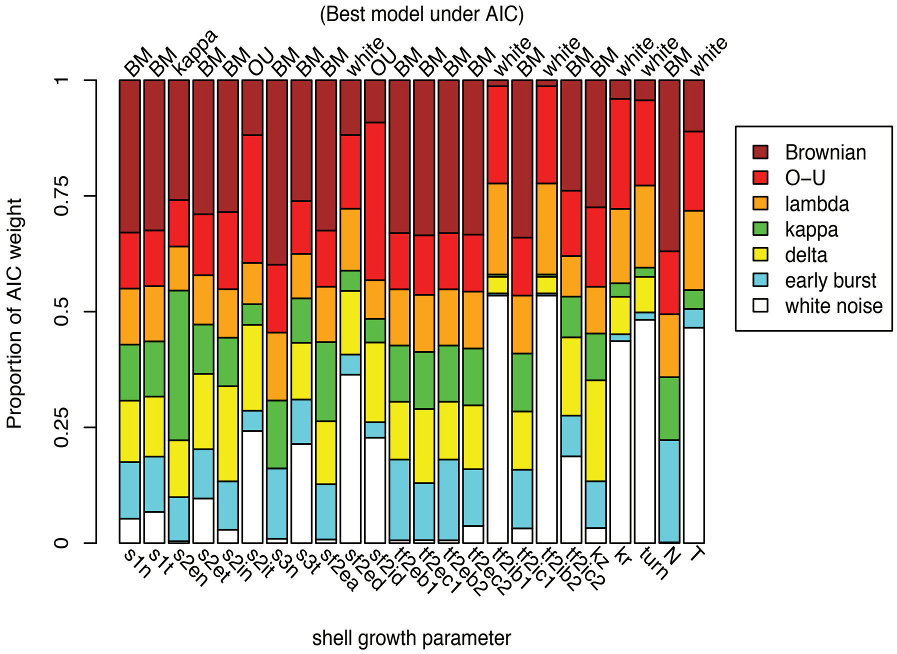


Figure S8. Akaike Information Criterion weights of 7 models for the evolution of neural network parameters. The models compared are: Brownian motion (BM, brown); the Ornstein−Uhlenbeck (OU, red) stabilizing selection model; the lambda model (orange) which rescales internal branch lengths by a linear fraction; the kappa model (green) which rescales each branch length by a power equal to the kappa parameter, and which becomes a speciational model as kappa approaches 0; delta (yellow) which focuses change towards the base or tips; early burst (EB, cyan) which has an initial high rate of change that then declines; and white noise (white), where observations are produced by a normal distribution with no tree structure, which represents the situation of no phylogenetic signal. Brownian motion (BM) has the highest AIC weight for 68% of the parameters (13/19). White noise (no phylogenetic signal) is superior for 16% (3/24) of the parameters. Parameter abbreviations on the x−axis correspond to those in Table S9, with underscores removed and with the last character being the Latin equivalent of the relevant Greek symbol.

Akaike Information Criterion. All of the models can be compared to each other at once using AIC weights (29), shown in Figure S8. Here, Brownian motion has the heaviest weight for 13/19 parameters. Typically this is a plurality rather than a majority weight, which is not surprising considering that the weight is being apportioned among seven models, the limited number of taxa limits the power to distinguish models, and the fact that the additional parameter of the non-Brownian models often converges upon parameter value that produces the Brownian model.

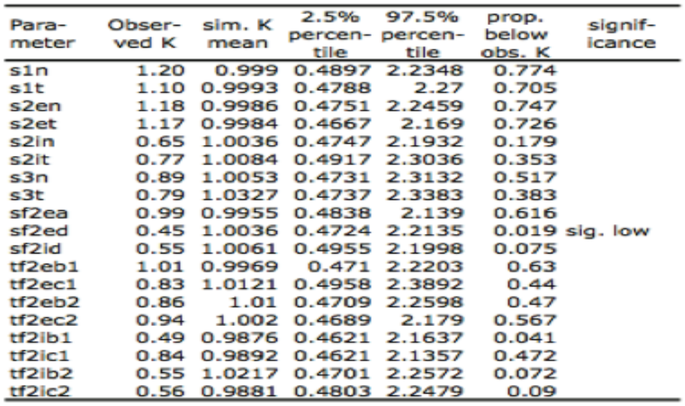
Non-parametric tests. Blomberg *et al*.'s (32) K is a measure of phylogenetic signal. K is the ratio of the observed MSE_0/MSE and expected MSE_0/MSE, where MSE_0 is the mean squared error between the phylogenetically correct mean and the tip data, and MSE is the mean squared error derived from the variance-covariance matrix calculated from the phylogenetic tree. The observed MSE_0/MSE is calculated from the data, and the expected MSE_0/MSE is calculated from 1000 nulls generated by reshuffling the tip data. K=1 indicates that a trait is evolving via Brownian motion and that trait data has a variance-covariance matrix that mirrors the phylogenetic structure. K<1 indicates less phylogenetic signal than expected under Brownian motion; K > 1 indicates phylogenetic overdispersion in the trait data (28).

Parameter	Blomberg's K	mean null K	2.5% null	97.5% null	p-value	signifi-cance
s1n	1.20	0.50	0.27	1.00	0.99	sig_high
s1t	1.10	0.48	0.25	0.87	0.99	sig_high
s2en	1.18	0.47	0.28	0.76	1.00	sig_high
s2et	1.17	0.48	0.25	0.91	0.98	sig_high
s2in	0.65	0.48	0.24	0.87	0.89	
s2it	0.77	0.52	0.21	1.48	0.92	
s3n	0.89	0.48	0.26	0.85	0.99	sig_high
s3t	0.79	0.48	0.26	0.91	0.95	
sf2ea	0.99	0.48	0.24	0.86	0.97	
sf2ed	0.45	0.58	0.18	3.14	0.55	
sf2id	0.55	0.49	0.23	1.06	0.75	
tf2eb1	1.01	0.48	0.25	0.85	0.95	
tf2ec1	0.83	0.48	0.24	0.93	0.98	sig_high
tf2eb2	0.86	0.48	0.25	0.88	0.94	
tf2ec2	0.94	0.48	0.25	0.89	0.96	
tf2ib1	0.49	0.55	0.18	2.91	0.64	
tf2ic1	0.84	0.48	0.27	0.86	0.99	sig_high
tf2ib2	0.55	0.60	0.18	3.30	0.76	
tf2ic2	0.56	0.48	0.25	0.84	0.76	
kz	1.11	0.50	0.26	1.05	0.95	
kr	0.46	0.48	0.24	0.85	0.55	
turn	0.52	0.48	0.26	0.79	0.67	
N	1.12	0.52	0.24	1.32	0.95	
T	0.61	0.62	0.18	3.68	0.85	

Table S7. Observed values of Blomberg's K for each neural network parameter, compared to a distribution generated under a null hypothesis of no phylogenetic signal (tip data randomly reshuffled 1000 times). The proportion of the null distribution beneath each observed K is reported; all proportions are above 0.5; 7 are significant at the 0.025 level. Parameter abbreviations as in previous tables.

The K statistics calculated for observed trait data were compared to two null hypotheses, one generating K statistics under a null hypothesis of no phylogenetic signal (tip data reshuffled 1000 times), and a second generating K statistics by simulating data under the Brownian motion fit to each trait. The proportion of the null distribution less than the observed K was calculated for each paramter. Under the no-signal null hypothesis, all observed K statistics are in the top 50% of the null distribution, indicating that the observed K is always higher than the mean of the null distribution. 13/19 observed Ks are in the top 10% of their nulls, 11/19 in the top 5%, and 7/19 in the top 2.5%. This indicates that most traits exhibit more phylogenetic signal than expected by chance (Table S7).

Comparison of the observed K statistic to K statistics generated by simulating under the best-fit Brownian model for each trait allow the detection of traits that show significantly more or less phylogenetic signal than expected under Brownian motion. Results are presented in . Only one parameter has a significantly lower K (and thus, less phylogenetic signal) than is expected under this null distribution (SF_2_e, σ_e, p=0.019), although the 95% confidence interval is quite broad and two other parameters would have significantly low Ks in a one-tailed test (Table S8).



Parameter	Observed K	sim. K mean	2.5% percentile	97.5% percentile	prop. below obs. K	significance
s1n	1.20	0.999	0.4897	2.2348	0.774	
s1t	1.10	0.9993	0.4788	2.27	0.705	
s2en	1.18	0.9986	0.4751	2.2459	0.747	
s2et	1.17	0.9984	0.4667	2.169	0.726	
s2in	0.65	1.0036	0.4747	2.1932	0.179	
s2it	0.77	1.0084	0.4917	2.3036	0.353	
s3n	0.89	1.0053	0.4731	2.3132	0.517	
s3t	0.79	1.0327	0.4737	2.3383	0.383	
sf2ea	0.99	0.9955	0.4838	2.139	0.616	
sf2ed	0.45	1.0036	0.4724	2.2135	0.019	sig. low
sf2id	0.55	1.0061	0.4955	2.1998	0.075	
tf2eb1	1.01	0.9969	0.471	2.2203	0.63	
tf2ec1	0.83	1.0121	0.4958	2.3892	0.44	
tf2eb2	0.86	1.01	0.4709	2.2598	0.47	
tf2ec2	0.94	1.002	0.4689	2.179	0.567	
tf2ib1	0.49	0.9876	0.4621	2.1637	0.041	
tf2ic1	0.84	0.9892	0.4621	2.1357	0.472	
tf2ib2	0.55	1.0217	0.4701	2.2572	0.072	
tf2ic2	0.56	0.9881	0.4803	2.2479	0.09	

Table S8. Observed values of Blomberg's K for each neural network parameter, compared to a distribution generated by simulating each trait under a pure Brownian motion model, at the rate estimated by ML for each parameter. The proportion of the null distribution beneath the observed K statistic is reported; Brownian motion can be rejected in only 1 case at p<0.025 level (two-tailed non-parametric test). Parameter abbreviations as in previous tables.

As one additional check that the Brownian motion assumption was reasonable to use for PIC and ancestral state reconstruction, a variety of diagnostic plots were generated as suggested by (30) and D. Ackerly (personal communication). These confirmed that in general the data were roughly normally distributed, and that the PICs did not show significant correlations with node depth or node value.

Brownian motion models are often run on ln-transformed data, e.g. if the trait distribution is highly skewed or varies over several orders of magnitude (as can happen with e.g. body size or genome size), or if there is concern that having a lower trait boundary of 0 could violate the assumptions of Brownian motion. Therefore, all of the above tests, and subsequent ancestral state reconstruction, were also replicated on a ln-transformed version of the neural net parameters. (For a few parameters, this required setting 0 values to 10% of the minimum nonzero value for that parameter, before ln-transformation.) However, this did not produce substantial change in results or interpretation and the results of ancestral state estimation were highly correlated, as might be expected given that the raw parameter data is roughly normally distributed and typically not close to 0.

9 Ancestral State Reconstruction

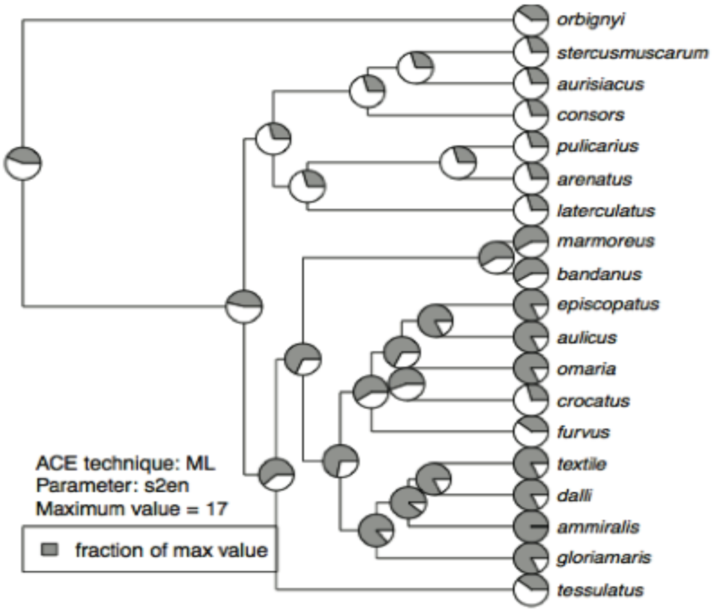


Figure S9. Maximum likelihood ancestral parameter estimates for neural net parameter S_2_e, ν (excitatory neuron response function, ν parameter; abbreviated s2en).

Ancestral states were estimated for each continuous parameter using maximum likelihood estimation under the Brownian motion model, as implemented in the ace function of the R package APE (16). After ancestral parameter values were estimated, shell patterns were generated from them using the same model as used for the living species and plotted on the phylogeny. Discrete characters were also estimated using ML; the color estimation was incorporated into the illustrations of the ancestors.

Results

ML estimation of continuous parameters. Examples of maximum likelihood estimations of ancestral parameter values are shown in Figure S9 and Figure S10.

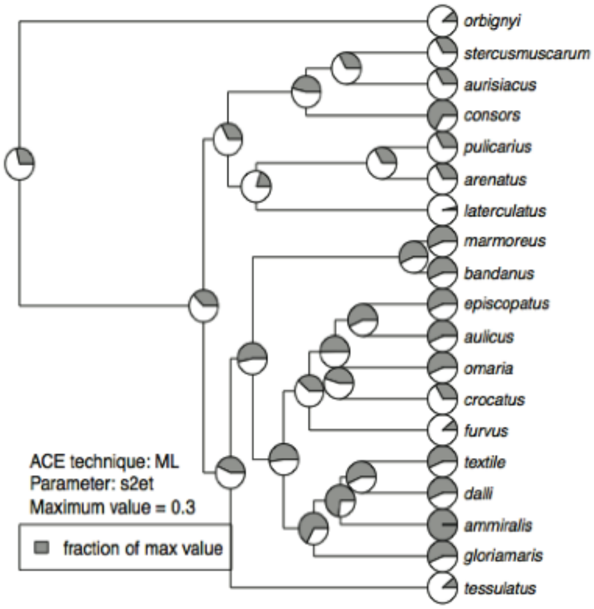


Figure S10. Maximum likelihood ancestral parameter estimates for neural net parameter S_2_e, θ (excitatory neuron response function, θ parameter; abbreviated s2et).

Inspection of the estimated history of these parameters and the others indicates that many clades do show similarities in parameter values with nearby species. However, as expected, estimates for each parameter tend towards the overall (phylogenetically corrected) average as estimates are made for ancestors further and further back in time, and uncertainty increases.

ML estimation of discrete characters.

In addition to the character mapping described in the main text, other discrete characters were also mapped for comparison (Figure S11). Cone shape is fairly scattered but shows some uniformity in small clades. Strikingly, prey preference shows extremely high conservation (as was clear in the discussion of Nam *et al.*) compared to shell pattern characters. Each major clade is almost completely restricted to a certain prey, and the entire pattern is explained by the minimum possible number of transitions.

Figure S11 also shows the distributions of dots and color (black versus orange/brown) in this group and the maximum likelihood assignment of ancestral states. The presence and absence of dots and cone shape are scattered throughout the phylogeny, indicating that they are evolutionarily labile, although orange/brown color shows some correlation with large clades.

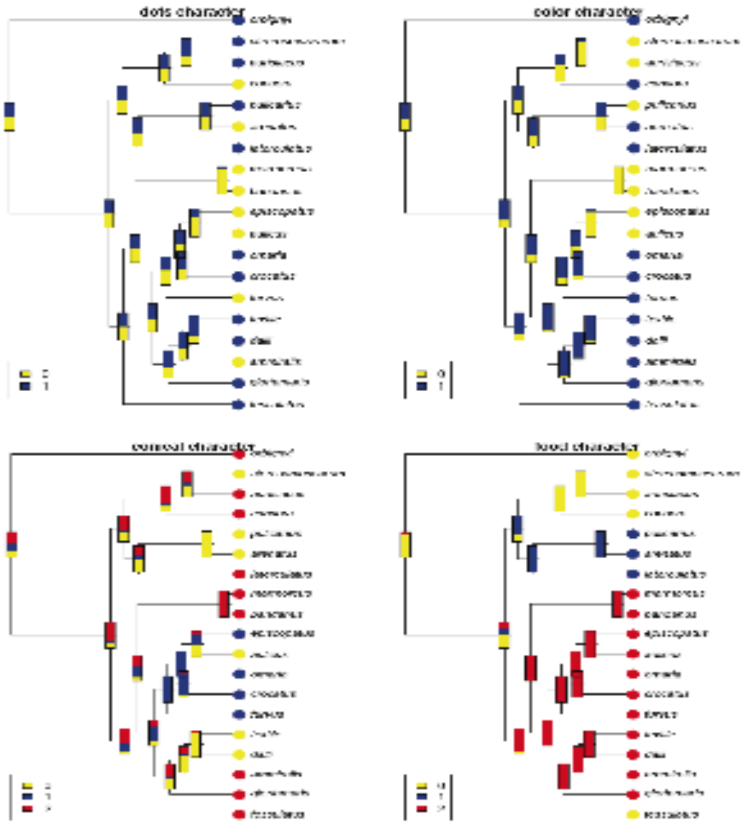


Figure S11. ML estimation of history of discrete characters. Character coding: dots 0/1 present/absent; color: 0=black and white, 1=orange/brown and white; conical: 0=rounded, 1=slightly rounded, 2=conical; food: 0=piscivorous, 1=vermivorous, 2=molluscivorous. See text for details.

References

1. Wilson H & Cowan J (1972) Excitatory and inhibitory interactions in localized populations of model neurons. *Biophysical Journal* 12:1-24.

2. Wilson H & Cowan J (1973) A Mathematical Theory of the Functional Dynamics of Cortical and Thalamic Nervous Tissue. *Kybernetik* 13:55-80.

3. Waddington C & Cowe R (1969) Computer simulation of a molluscan pigmentation pattern. *J. Theo. Biol* 25:219-225.

4. Kusch I & Markus M (1996) Mollusc Shell Pigmentation: Cellular automaton simulations and evidence for undecidability. *J. Theor. Biol.* 178:333-340.

5. Wolfram S (1984) Cellular automata as models of complexity. *Nature* 311:419-424.

6. Meinhardt H (1984) Models for positional signalling, the threefold subdivision of segments and the pigmentation pattern of molluscs. *J. Embryol. exp. Morphol.* 83 Suppl:289-311.

7. Meinhardt H & Klingler M (1986) *Pattern formation by coupled oscillations: the pigmentation patterns on the shells of molluses* (preprint).

8. Meinhardt H & Klingler M (1987) A model for pattern formation on the shells of molluscs. *J. theor. Biol* 126:63-89.

9. Meinhardt H, Prusinkiewicz P, & Fowler D (2003) *The Algorithmic Beauty of Sea Shells, 3rd ed.* (Springer, New York).

10. Fowler DR, Meinhardt H, & Prusinkiewicz P (1992) Modeling seashells. *Computer Graphics* 26:379-387.

11. Murray JD (2003) *Mathematical biology* (Springer, New York) 3rd Ed pp xxv, 811.

12. Campbell J, Ermentrout B, & Oster G (1986) A model for mollusk shell patterns based on neural activity. *The Veliger* 28:369-388.

13. Boettiger A, Ermentrout B, & Oster G (2009) The neural origins of shell structure and pattern in aquatic mollusks. *PNAS* 106:6837-6842.

14. Ihaka R & Gentleman R (1996) R: a language for data analysis and graphics. *Journal of computational and graphical statistics* 5(3):299-314.

15. R Team (2010) *R: A language and environment for statistical computing (Version 2.12.1)* (R Foundation for Statistical Computing, Vienna, Austria).

16. Paradis E (2006) Analysis of Phylogenetics and Evolution with R.

17. Harmon LJ, Weir JT, Brock CD, Glor RE, & Challenger W (2008) GEIGER: investigating evolutionary radiations. *Bioinformatics* 24(1):129-131.

18. Schliep KP (2010) Phylogenetic analysis in R. *Bioinformatics*.

19. Nam HH, Corneli PS, Watkins M, Olivera B, & Bandyopadhyay P (2009) Multiple genes elucidate the evolution of venomous snail-hunting Conus species. *Molecular Phylogenetics and Evolution* 53(3):645-652.

20. Sanderson MJ (1997) A nonparametric approach to estimating divergence times in the absence of rate constancy. *Molecular Biology and Evolution* 14(12):1218-1231.

21. Robinson DF & Foulds LR (1981) Comparison of phylogenetic trees. *Mathematical Biosciences* 53(1-2):131-147.

22. Penny D & Hendy MD (1985) The use of tree comparison metrics. *Systematic Zoology* 34(1):75-82.

23. Rzhetsky A & Nei M (1992) A simple method for estimating and testing minimum-evolution trees. *Mol. Biol. Evol* 9(5):945-967.

24. Billera LJ, Holmes SP, & Vogtmann K (2001) Geometry of the Space of Phylogenetic Trees. *Advances in Applied Mathematics* 27(4):733-767.

25. Felsenstein J (1985) Phylogenies and the comparative method. *American Naturalist* 125(1):1-15.

26. Felsenstein J (1988) Phylogenies and Quantitative Characters. *Annual Review of Ecology and Systematics* 19:445-471.

27. Pagel M (1999) Inferring the historical patterns of biological evolution. *Nature* 401(6756):877-884.

28. Ackerly D (2009) Conservatism and diversification of plant functional traits: Evolutionary rates versus phylogenetic signal. *Proceedings of the National Academy of Sciences* 106(Supplement 2):19699-19706.

29. Harmon LJ*, et al.* (2010) Early bursts of body size and shape evolution are rare in comparative data. *Evolution* 64(8):2385-2396.

30. Blomberg SP, Theodore Garland JR, & Ives AR (2003) Testing for phylogenetic signal in comparative data: behavioral traits are more labile. *Evolution* 57(4):717-745.

31. Felsenstein J (2004) Inferring phylogenies.xx, 664 p.

32. Anderson JC*, et al.* (2003) Modular organization of adaptive colouration in flounder and cuttlefish revealed by independent component analysis. *Network: Computation in Neural Systems* 14(2):321-333.

-