Mathematical Neuroscience - Assignment 3

Neuronal coupling and Neural Fields

Matuš Halák - 2724858 (VU Student ID)

May 2025

1 Coupled Neurons

1.1 Firing time map and Δ for non-identical coupled neurons

For this question, we use equation 8.2 from Ermentrout and Terman defining a general PRC $\Delta(\phi) = T - T'$ where T is the period of the neuron when firing in isolation and T' is the time difference between the current spike and previous spike. For a PRC for two nonidentical oscillating neurons we adapt this to:

$$\Delta_i(\phi_j) = T_i - T_i'(\phi_j) \quad \text{for } i \in (1,2) \text{ and } j \in (2,1) \tag{1}$$

Where now $T_i'(\phi_j)$ indicates that the time difference between current spike and previous spike is determined by phase ϕ_j of the input j from the coupled neuron.

To derive the firing map for non-identical coupled neurons where $T_1 = T_0$ and $T_2 = T_0 + \varepsilon$, we adapt the derivation from section 8.2.3.1 from Ermentrout and Terman as follows:

$$\begin{aligned} t_1' &= t_1 + T_0 - \Delta_1(t_2 - t_1) \\ t_2' &= t_2 + T_0 + \varepsilon - \Delta_2(t_1' - t_2) \\ \tau &= t_2 - t_1; \qquad \tau' = t_2' - t_1' \end{aligned}$$

After subtracting t_2 from both sides of the t_1' equation and substituting in τ and τ':

$$\begin{aligned} t_1' - t_2 &= -(t_2 - t_1) + T_0 - \Delta_1(\tau) = -\tau + T_0 - \Delta_1(\tau) = A \\ t_2' &= t_2 + T_0 + \varepsilon - \Delta_2(A) \end{aligned}$$

Following Ermentrout and Terman we subtract A from both sides of the t_2' equation:

$$\begin{aligned} t_2' - (t_1' - t_2) &= t_2 + T_0 + \varepsilon - \Delta_2(A) - A \\ \tau' &= T_0 + \varepsilon - \Delta_2(A) + \tau - T_0 + \Delta_1(\tau) \end{aligned}$$

And thus for $\tau_{n+1} = \tau'$ and $\tau_n = \tau$ we obtain the firing time map $M(\tau_n)$ which now also includes the phase difference ε between the two neurons and two distinct PRC's Δ_1 and Δ_2:

$$\tau_{n+1} = \tau_n + \varepsilon + \Delta_1(\tau_n) - \Delta_2(T_0 - \tau_n - \Delta_1(\tau_n)) = M(\tau_n) \tag{2}$$

1.2 Fixed points of coupled neurons with period difference ε

Now we investigate a case where $\Delta_1(\phi) = \Delta_2(\phi) = a\sin(2\pi\phi)$ with coupling strength between the two neurons $a \in (-1, 0)$. To study fixed points of $M(\tau)$, we note that fixed points represent phase-locked

solutions where τ does not change over time:

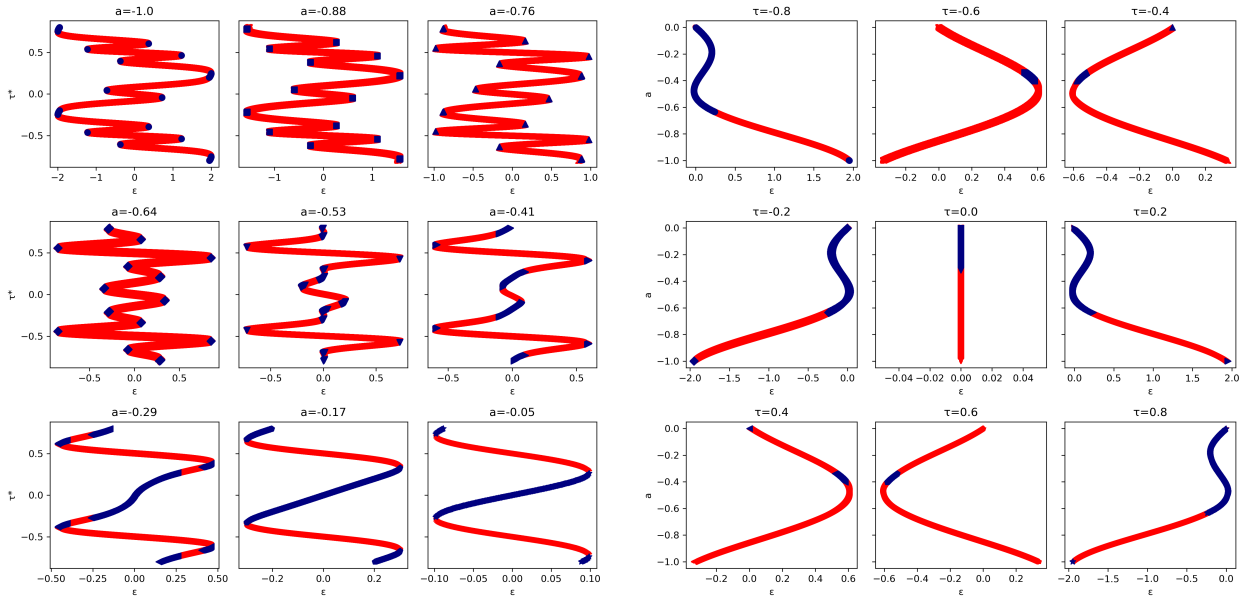
$$\tau_* = \tau_{n+1} = \tau_n = M(\tau_n)$$
$$M(\tau_n) - \tau_* = 0$$
$$\varepsilon + \Delta_1(\tau_n) - \Delta_2(T_0 - \tau_n - \Delta_1(\tau_n)) = 0$$

And we therefore have an analytical solution for the fixed points of $M(\tau)$:

$$\varepsilon_* = \Delta_2(T_0 - \tau_n - \Delta_1(\tau_n)) - \Delta_1(\tau_n) \tag{3}$$

For numerical investigation, because $\Delta(\phi)$ can also be seen as $\Delta(\phi, a)$ if we treat a as a parameter, and Equation 3 is a function of ε and τ, we can reformulate the fixed points equation for ε_* as a function of τ and a. To remove T_0 as a variable, we set $T_0 = 1$ and get:

$$\varepsilon_*(\tau, a) = \Delta_2(1 - \tau - \Delta_1(\tau, a), a) - \Delta_1(\tau, a) \tag{4}$$



(a) Period difference (ε_*) vs phase offset (τ_*) across a (b) Period difference ε_* vs coupling strength a across τ

Figure 1: Fixed points of two coupled neurons in the (ε, τ, a) space. Unstable (red), Stable (blue)

Figure 1 shows the results of our numerical investigation in Python across a range of ε, τ and a. While Figure 1b was obtained using Equation 3, Figure 1a was obtained with root finding for $M(\tau)$ (Equation 2). Since subfigures in 1 present different slices of the (ε, τ, a) space, they provide complementary perspective on the nature of fixed points for two non-identical coupled neurons. We classify fixed points as attracting / stable if $M'(\varepsilon_*, \tau_*, a_*) < 1$, because it means that around that point, phase differences are attracted and decay towards 0 (synchrony). Conversely, $M'(\varepsilon_*, \tau_*, a_*) > 1$ are unstable fixed points, because around them, phase differences grow over time.

The results reveal that for strongly coupled neurons (large $|a|$) there exist many unstable branches of fixed points across phase offsets τ, which "turn" when unstable (red) and stable (blue) branches meet at saddle-node bifurcations at different values of ε beyond which there are no more fixed points for that phase offset and that coupling strength. This can be best seen in Figure 1a, where we applied a fine τ grid at various coupling strengths, while most of those branch turns (saddle-node bifurcations) at high coupling

strengths cannot be seen in Figure 1b because there we only show a few phase offsets. Notably, at these high coupling strengths, many different rhythms across a wide range of period differences e can co-exist (e.g., at $|a| = 1$ fixed points exist for $\varepsilon \in (-2, 2)$). As we decrease the coupling strength $|a|$, we note the following developments. Firstly, the range of period differences that can coexist starts to shrink from $\varepsilon \in (-2, 2)$ to $\varepsilon \in (-1, 1)$ to $\varepsilon \in (-0.5, 0.5)$ to $\varepsilon \in (-0.1, 0.1)$ and finally at $a = 0$ there are no longer any fixed points since the neurons are uncoupled. Secondly, this reduction in supported range of period differences with decreasing coupling strength goes together with an increasing prominence of synchronous branches. In other words, at weak $|a|$ neurons fall more and more into synchrony, as the unstable branches supported at stronger coupling collide and disappear, leaving behind the stable fixed points. As more and more of these collisions happen as the ε range is shrunk, the stable fixed points start to join into a one big stable branch which dominates at low coupling strengths. Therefore, the main finding is that strongly coupled neurons can show many different rhythms across a wider range of possible period differences, whereas weakly coupled neuron are forced closer to synchrony at a much narrower range of possible period differences.

2 Neural Field Analysis

In the following section a neural field model defined as:

$$\partial_t u(x,t) = -u(x,t) + \int_{\mathbf{R}} w(x-y) f(u(y,t) - h) dy, \qquad (x,t) \in \mathbf{R} \times \mathbf{R}_{>0} \tag{5}$$

$$u(x,0) = u_0(x), \qquad x \in \mathbf{R} \tag{6}$$

Where the synaptic kernel $w(z)$ is the "Mexican hat" kernel and the firing rate function $f(u)$ is the Heaviside function $H(u)$:

$$w(z) = (1 - |z|)e^{-|z|} \tag{7}$$

$$f(u) = H(u) = H(u - h) = 1_{u \geq h} \qquad \text{for firing threshold } h \tag{8}$$

In the following sections, we are able to analyze this Neural Field model using Amari's approach because we use the Heaviside firing rate function.

2.1 Steady state $U(x)$

2.1.1 $U(x)$ between x_1 and x_2

First, since the bump $U(x)$ crosses the threshold h twice, this means that at the crossing points, $u(x,t) = h$. Second, we identify that, in the steady state, $\partial_t u(x,t) = 0$ and since it is a steady state that does not change over time, as $t \to \infty$, $u(x,t) = U(x)$. We also insert the Heaviside firing rate function 8 and note it's properties around x_1 and x_2:

$$\begin{aligned} u(x_1,t) = u(x_2,t) = h \quad \forall t \\ U(x) = \int_{\mathbf{R}} w(x-y) H(u(y,t) - h) dy \\ H(u(y,t) - h) = 0 \quad \forall (y < x_1 \vee y > x_2) \\ H(u(y,t) - h) = 1 \quad \forall (y \geq x_1 \vee y \leq x_2) \end{aligned}$$

Because $H(u(x,t) - h)$ is 0 everywhere except between x_1 and x_2, where it is 1, it also means that the integral is 0 in all areas except between x_1 and x_2, and between x_1 and x_2, the Heaviside acts just like an indicator function, leaving us with:

$$U(x) = \int_{x_1}^{x_2} w(x-y) dy \tag{9}$$

2.1.2 Solution for $U(x)$

First we express $U(x)$ with the full synaptic kernel 7:

$$U(x) = \int_{x_1}^{x_2} (1 - |x - y|)e^{-|x-y|}dy$$

Following the hint, we first show for a generic $x \in \mathbf{R}$ that $\int_a^x w(x-y)dy = (x-a)e^{-|x-a|}$:

$$\begin{aligned}
I(x,a) &= \int_a^x (1-|x-y|)e^{-|x-y|}dy \quad \Big| \quad t = x-y, dy = -dt, \\
&= -\int_{x-a}^{0} (1-|t|)e^{-|t|}dy = \int_0^{x-a} (1-|t|)e^{-|t|}dy \quad \Big| \quad B = x-a \\
I(B) &= \int_0^B (1-|t|)e^{-|t|}dy \qquad \text{sgn(B) does not change between 0 and B} \\
&= \begin{cases} \int_0^B (1-t)e^{-t}dy, & B \geq 0 \\ \int_0^B (1+t)e^{t}dy, & B < 0 \end{cases} \qquad \Big| \text{ Wolfram alpha to solve the integrals for each case} \\
&= \begin{cases} Be^{-B} = Be^{-|B|}, & B \geq 0 \\ Be^{B} = Be^{-(-B)} = Be^{-|B|}, & B < 0 \end{cases} \\
I(B) &= Be^{-|B|} \quad \Big| \quad \text{substitute back } B = x-a \\
I(x,a) &= (x-a)e^{-|x-a|}
\end{aligned}$$

We then split the integral into $\int_{x1}^{x2} = \int_{x_1}^{x} + \int_{x}^{x_2}$ for $x \in \mathbf{R}$ and obtain:

$$\int_{x_1}^{x} w(x-y)dy = (x-x_1)e^{-|x-x_1|} \quad \text{and} \quad \int_{x}^{x_2} w(x-y)dy = (x_2-x)e^{-|x_2-x|}$$

$$\int_{x1}^{x2} w(x-y)dy = (x-x_1)e^{-|x-x_1|} + (x_2-x)e^{-|x_2-x|}$$

And so because $U(x) = \int_{x1}^{x2} w(x-y)dy = \int_{x_1}^{x} w(x-y)dy + \int_{x}^{x_2} w(x-y)dy$, the solution for $U(x)$ is:

$$U(x) = (x-x_1)e^{-|x-x_1|} + (x_2-x)e^{-|x_2-x|} \tag{10}$$

We can reformulate this in terms of $\phi(\Delta)$ and get:

$$U(x) = \phi(x-x_1) + \phi(x_2-x), \qquad \phi(\Delta) = \Delta e^{-|\Delta|}. \tag{11}$$

This is equivalent to the equation in 2.2 of the assignment where the sgn(Δ) is used and $|\Delta|$ is passed into ϕ which then does not take absolute value in the exponent. To me, that formulation seems unnecessary, so I proceed with Equation 11, which follows directly from the generic case which I derived above.

2.1.3 Properties of localized bump $U(x)$

First we verify that $\lim_{x\to\pm\infty} U(x) = 0$:

$$\begin{aligned}
\lim_{x\to\pm\infty} U(x) &= \lim_{x\to\pm\infty} (x-x_1)e^{-|x-x_1|} + (x_2-x)e^{-|x_2-x|} \\
&= \lim_{x\to\pm\infty} (x-x_1)e^{-|x-x_1|} + \lim_{x\to\pm\infty} (x_2-x)e^{-|x_2-x|} \\
&= (x-x_1)\lim_{x\to\pm\infty} e^{-|x-x_1|} + (x_2-x)\lim_{x\to\pm\infty} e^{-|x_2-x|} \\
&= \lim_{x\to\pm\infty} e^{-|x-x_1|} + \lim_{x\to\pm\infty} e^{-|x_2-x|} \\
&= \lim_{x\to\pm\infty} e^{-|x|} + \lim_{x\to\pm\infty} e^{-|-x|} \\
&= \lim_{x\to\pm\infty} \frac{1}{e^{|x|}} + \lim_{x\to\pm\infty} \frac{1}{e^{|x|}} \\
\lim_{x\to\pm\infty} U(x) &= 0+0=0
\end{aligned}$$

To show that $U(x) \in C^1(\mathbf{R})$ I tried to do the following [1]. I found that $C^1(\mathbf{R})$ functions must meet three criteria:

1. Function must be continuous. This could seem like and issue because we did use a discontinuous Heaviside function to derive U(x). For this criterion, we check if the left and right limit are the same around the critical values x_1 and x_2 where the discontinuity happens in $H(u(x))$. If we use Equation 11 we see that:

$$\begin{aligned}
U(x_1) &= \phi(0) + \phi(x_2 - x_1) \\
U(x_2) &= \phi(x_2 - x_1) + \phi(0) \\
\phi(0) = 0e^0 = 0 \quad &\text{and} \quad \phi(x_2-x_1) = (x_2-x_1)e^{-|x_2-x_1|}
\end{aligned}$$

So at both the critical values $U(\text{critical}) = \phi(0) + \phi(x_2 - x_1)$ where critical $\in (x_1, x_2)$, and both $\phi(0)$ and $\phi(x_2 - x_1)$ are continous:

$$\begin{aligned}
\lim_{\Delta\to 0^+} \phi(\Delta) &= \lim_{\Delta\to 0^-} \phi(\Delta) = 0 \\
\lim_{\Delta\to (x_2-x_1)^+} \phi(\Delta) &= \lim_{\Delta\to (x_2-x_1)^-} \phi(\Delta) = (x_2-x_1)e^{-|x_2-x_1|}
\end{aligned}$$

2. Function must be differentiable. For this, since $U(x)$ is a sum of two $\phi(\Delta)$'s, we check whether $\phi(\Delta)$ is differentiable:

$$\begin{aligned}
\phi(\Delta) &= \Delta e^{-|\Delta|} \\
\phi(\Delta) &= \begin{cases} \Delta e^{-\Delta}, & \text{if } \Delta \geq 0 \\ \Delta e^{\Delta}, & \text{if } \Delta < 0 \end{cases} \qquad \Big| \frac{d}{d\Delta}
\end{aligned}$$

And we find that not only is $\phi(\Delta)$ differentiable, but that $\phi'(\Delta) = w(\Delta)$:

$$\phi'(\Delta) = \begin{cases} e^{-\Delta} - \Delta e^{-\Delta} = (1-\Delta)e^{-\Delta}, & \text{if } \Delta \geq 0 \\ e^{\Delta} + \Delta e^{\Delta} = (1+\Delta)e^{-(-\Delta)}, & \text{if } \Delta < 0 \end{cases} \quad = (1-|\Delta|)e^{-|\Delta|} = w(\Delta) \tag{12}$$

$$U'(x) = \phi'(x-x_1) + \phi'(x_2-x) = w(x-x_1) + w(x_2-x) \tag{13}$$

[1] Please note that I had no idea what $C^1(\mathbf{R})$ functions are, but I tried my best based on what I learned online.

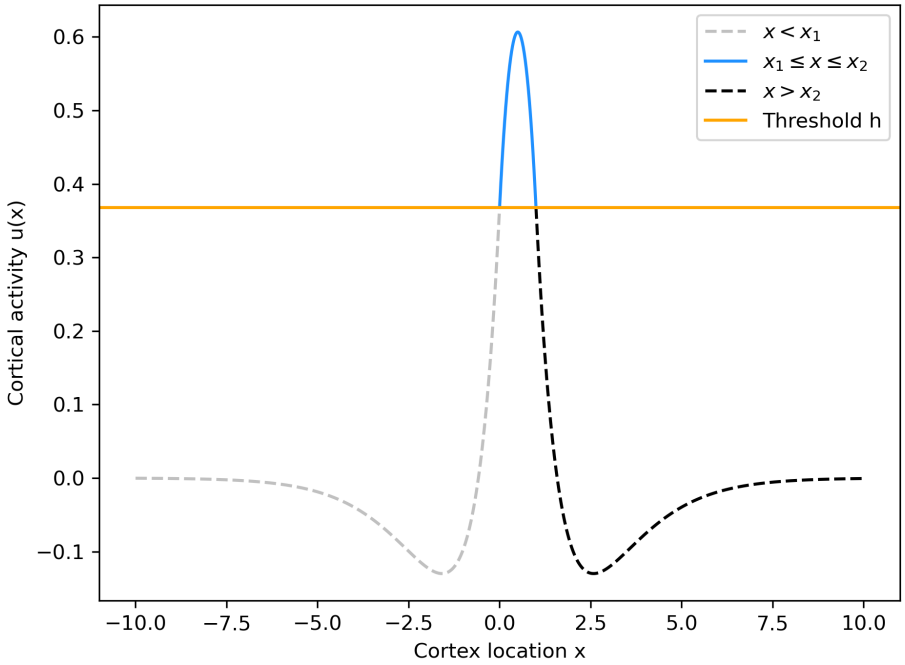


Figure 2: Localized bump $U(x)$ with bump width $\Delta = 1$. $(x_1, x_2) = (0, 1)$.

3. The derivative of the function must be continuous. Since we found above that $\phi'(\Delta) = w(\Delta)$, we only note that $w(\Delta)$ is continuous at both critical cases with:

$$\lim_{\Delta \to 0^+} w(\Delta) = \lim_{\Delta \to 0^-} w(\Delta) = (1-0)e^0 = 1$$
$$\lim_{\Delta \to (x_2-x_1)^+} w(\Delta) = \lim_{\Delta \to (x_2-x_1)^-} w(\Delta) = (1-|x_2-x_1|)e^{-|x_2-x_1|}$$

Therefore, $U(x)$ meets all three conditions and thus, $U(x) \in C^1(\mathbf{R})$.

Plotting a localized bump $U(x)$ for $x_1 = 0$ and $x_2 = 1$ (estimating h as the activity $u(x)$ that intersects $U(x)$ at x_1 and x_2), we get the following Figure 2, showing a bump localized around x_1 and x_2.

2.2 Bump width Δ and threshold h

We first define bump width as $\Delta = x_2 - x_1$. Because the bump crosses the threshold h twice, at x_1 and x_2 and $x_1 < x_2$, we can say that $U(x_1) = U(x_2) = h$. Using equation 9 and its property of translational invariance, we get an expression for U(x) in terms of the bump width Δ when we shift the integral by $-x_1$:

$$U(x) = \int_0^{x_2-x_1} w(x-y)dy = \int_0^{\Delta} w(x-y)dy$$

In equation 12, we see that $w(z) = \phi'(z)$ and therefore the anti-derivative of $w(z)$ is $\phi(z)$. Using Fundamental Theorem of Calculus we get:

$$U(x) = \int_0^{\Delta} w(z)dz = \phi(z)\Big|_{z=0}^{z=\Delta} = \phi(\Delta) - \phi(0)$$

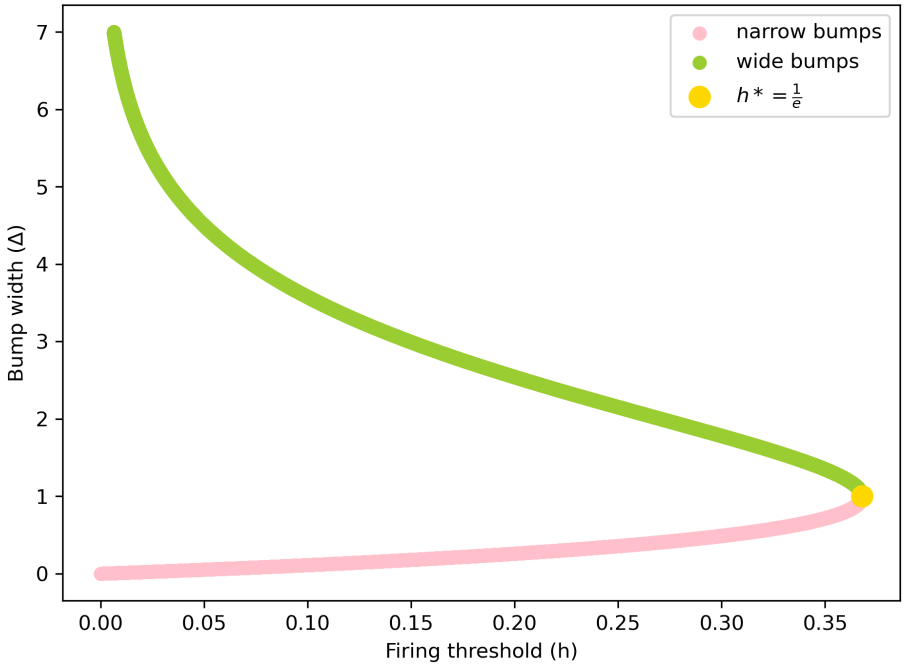


Figure 3: Bifurcation diagram for bump solutions

In this shifted version of the problem, we now have $x_1 = 0$ and $x_2 = \Delta$. As shown in previous exercise $\phi(0) = 0$, and because $U(x_1) = U(0) = h$ we find:

$$U(0) = h = \phi(\Delta) = \Delta e^{-|\Delta|} \tag{14}$$

Given that $w(\Delta)$ is the derivative of $\phi(\Delta)$, we look for the stationary point of $\phi(\Delta)$ by setting its derivative $w(\Delta) = 0$:

$$\begin{aligned} 0 = (1 - |\Delta|)e^{-|\Delta|} &= e^{-|\Delta|} - |\Delta|e^{-|\Delta|} \\ |\Delta|e^{-|\Delta|} &= e^{-|\Delta|} \\ \Delta_{stationary} &= 1 \end{aligned}$$

To check the nature of $\phi(\Delta = 1)$, we look at the sign of $w(\Delta)$ around 1:

$$w(\Delta) = (1 - |\Delta|)(\frac{1}{e^{|\Delta|}}) = (T_1)(T_2)$$

$$w(\Delta) = \begin{cases} (T_1 > 0)(T_2 > 0) \rightarrow w(\Delta) > 0 & \text{if } \Delta < 1 \\ (T_1 < 0)(T_2 > 0) \rightarrow w(\Delta) < 0 & \text{if } \Delta > 1 \end{cases}$$

From which we see that $\Delta = 1$ is a maximum of $\phi(\Delta)$ because at values lower than 1, $\phi'(\Delta) > 0$ (increasing), and at values higher than 1, $\phi'(\Delta) < 0$ (decreasing). Therefore, $\phi(1) = \frac{1}{e}$ is the maximum value $\phi(\Delta)$ can return. As such, because $h = \phi(\Delta)$, if $h > 1/e$, the threshold line h would lie above $\phi(\Delta)$ without intersecting it, meaning that there can be no solution where $U(x) \geq h$ and therefore no bumps exist. On the other hand, if $h < 1/e$, the threshold line h intersects $\phi(\Delta)$ twice (both as $\phi(\Delta)$ is rising towards its peak and as it is decreasing from the peak).

In Figure 3 we can see the bifurcation diagram for bump solutions obtained using custom-written Python script. We see two branches of solutions, one branch with narrow bump widths $\Delta < 1$ (pink) and a branch

with wide bump widths $\Delta > 1$ (green). The branches meet at $(h*, \Delta) = (1/e, 1)$, beyond which, as we showed (but as can also be seen in the diagram), there are no more bumps of any width. In fact, our previous Figure 2 shows exactly this edge case when $(h, \Delta) = (1/e, 1)$.

2.3 Finding the neural field eigenvalue problem

Here we show how to derive the neural field eigenvalue problem which will then help us study the stability of bumps. We start with the definition of the Neural Field 5 and want to study how the system behaves around equilibrium functions $U(x)$ that we have found if we perturb it slightly by $\tilde{u}(x,t) = e^{\lambda t}v(x)$ where $(\lambda, v(x))$ is an eigenpair of the neural field. Therefore, we try to derive an eigenproblem-like equation from:

$$\partial_t[U(x) + \tilde{u}(x,t)] = -[U(x) + \tilde{u}(x,t)] + \int_{\mathbf{R}} w(x-y) f(U(y) - h + \tilde{u}(y,t))dy \tag{15}$$

Because $U(x)$ is steady state and independent of time $\partial_t(U(x)) = 0$. Furthermore, we perform a Taylor expansion on the firing rate function inside the integral:

$$f(U(y) - h + \tilde{u}(y,t)) \approx f(U(y) - h) + f'(U(y) - h)\tilde{u}(y,t) + \mathcal{O}(\tilde{u}^2)$$

After discarding the higher order terms (will be minuscule given that $\tilde{u}$ is already small), our original equation becomes:

$$\begin{aligned}
\partial_t(\tilde{u}(x,t)) &= -U(x) - \tilde{u}(x,t) + \int_{\mathbf{R}} w(x-y)[f(U(y) - h) + f'(U(y) - h)\tilde{u}(y,t)]dy \\
&= -U(x) + \int_{\mathbf{R}} w(x-y) f(U(y) - h)dy - \tilde{u}(x,t) + \int_{\mathbf{R}} w(x-y) f'(U(y) - h)\tilde{u}(y,t)dy
\end{aligned}$$

Now because $U(x) = \int_{\mathbf{R}} w(x-y)H(u(y,t) - h)dy$, the 0th order term cancels out and we are left with the first order term in $\tilde{u}$:

$$\begin{aligned}
\partial_t(\tilde{u}(x,t)) &= -\tilde{u}(x,t) + \int_{\mathbf{R}} w(x-y) f'(U(y) - h)\tilde{u}(y,t)dy \\
\partial_t(e^{\lambda t}v(x)) &= -e^{\lambda t}v(x) + \int_{\mathbf{R}} w(x-y) f'(U(y) - h)e^{\lambda t}v(y)dy \quad \Big| \quad \text{differentiate LHS} \\
\lambda e^{\lambda t}v(x) &= -e^{\lambda t}v(x) + \int_{\mathbf{R}} w(x-y) f'(U(y) - h)e^{\lambda t}v(y)dy \quad \Big| \quad + e^{\lambda t}v(x); \text{bring } e^{\lambda t} \text{ in front of } \int \\
\lambda e^{\lambda t}v(x) + e^{\lambda t}v(x) &= e^{\lambda t}\int_{\mathbf{R}} w(x-y) f'(U(y) - h)v(y)dy \\
(\lambda + 1)e^{\lambda t}v(x) &= e^{\lambda t}\int_{\mathbf{R}} w(x-y) f'(U(y) - h)v(y)dy \quad \Big| \quad e^{\lambda t} \text{ terms cancel out} \\
(\lambda + 1)v(x) &= \int_{\mathbf{R}} w(x-y) f'(U(y) - h)v(y)dy
\end{aligned}$$

The final equation presents us with an issue because our $f(u)$ is the Heaviside function which is not differentiable at 0 and so we cannot simply use $H'(U(y) - h)$ instead of $f'(U(y) - h)$. However, we can approximate $H'(U(y) - h)$ using Dirac's delta (δ), which given an argument of the form $g(U(y) - h)$ returns the value of g when $U(y) - h = 0$. Therefore, even though $H'(U(y) - h)$ is not differentiable, we can treat $H'(U(y) - h)$ as any general $g(U(y) - h)$ and get output from the δ when $U(y) - h = 0$. We already know that there are two such cases, x_1 and x_2. Therefore, we can wrap the $H'(U(y) - h)$ inside the integral with δ like this[2]:

$$\delta(H'(U(y) - h)) = \sum_{j=1}^{2} \frac{\delta(y - x_j)}{|U'(x_j)|}$$

[2]This is really pushing the limits of what I can understand with limited background in Mathematics.

and ultimately get:

$$(\lambda+1)v(x) = \int_{\mathbf{R}} w(x-y)\sum_{j=1}^{2}\frac{\delta(y-x_j)}{|U'(x_j)|}v(y)dy \quad \Big| \quad \text{move } \sum \text{ in front of } \int$$

$$(\lambda+1)v(x) = \sum_{j=1}^{2}\int_{\mathbf{R}} w(x-y)\frac{\delta(y-x_j)}{|U'(x_j)|}v(y)dy$$

Because δ only picks up the integrand when $y = x_j$, we actually end up with simply:

$$(\lambda+1)v(x) = \sum_{j=1}^{2}\frac{w(x-x_j)}{|U'(x_j)|}v(x_j) \tag{16}$$

Which is the neural field eigenproblem we were looking for.

2.4 The eigenvalue problem

Having found the neural field eigenproblem 16, we turn towards trying to find the eigenvalues and eigenmodes / eigenfunctions of the neural field.

2.4.1 Matrix M

We have shown in equation 13 that $U'(x) = w(x-x_1) + w(x_2-x)$. However, here it becomes more useful to have subtraction rather than addition between the two terms of $U'(x)$. Because $w(z) = \phi'(z)$ and $\phi(z)$ has the property that $-\phi(z) = \phi(-z)$, and because $w(z)$ is even ($w(z) = w(-z)$) we can simply change our $U'(x)$ equation to an equivalent form :

$$U'(x) = w(x-x_1) + w(-(x_2-x)) = w(x-x_1) - w(x-x_2) \tag{17}$$

Substituting x_1 and x_2 into the eigenproblem we get:

$$(\lambda+1)v(x_1) = \frac{w(x_1-x_1)}{|U'(x_1)|}v(x_1) + \frac{w(x_1-x_2)}{|U'(x_2)|}v(x_2)$$
$$(\lambda+1)v(x_2) = \frac{w(x_2-x_1)}{|U'(x_1)|}v(x_1) + \frac{w(x_2-x_2)}{|U'(x_2)|}v(x_2)$$

Using $\Delta = x_2 - x_1$ and fully writing out the U' terms according to equation 17 and using the even property of $w(z)$:

$$U'(x_1) = w(0) - w(\Delta) \qquad \text{and} \qquad U'(x_2) = w(\Delta) - w(0) = -(w(0) - w(\Delta))$$
$$|U'(x_1)| = |U'(x_2)| = |w(0) - w(\Delta)|$$
$$(\lambda+1)v(x_1) = \frac{w(0)}{|w(0)-w(\Delta)|}v(x_1) + \frac{w(\Delta)}{|w(0)-w(\Delta)|}v(x_2) \quad \Big| -v(x_1)$$
$$(\lambda+1)v(x_2) = \frac{w(\Delta)}{|w(0)-w(\Delta)|}v(x_1) + \frac{w(0)}{|w(0)-w(\Delta)|}v(x_2) \quad \Big| -v(x_2)$$

$$\lambda v(x_1) = \left(\tfrac{w(0)}{|w(0)-w(\Delta)|} - 1, \tfrac{w(\Delta)}{|w(0)-w(\Delta)|}\right)\begin{pmatrix}v(x_1)\\v(x_2)\end{pmatrix}, \qquad \lambda v(x_2) = \left(\tfrac{w(\Delta)}{|w(0)-w(\Delta)|}, \tfrac{w(0)}{|w(0)-w(\Delta)|} - 1\right)\begin{pmatrix}v(x_1)\\v(x_2)\end{pmatrix}$$

Our final vector equations can be easily put into full matrix notation $\lambda v = Mv$ to produce:

$$\lambda\begin{pmatrix}v(x_1)\\v(x_2)\end{pmatrix} = \begin{pmatrix}\frac{w(0)}{|w(0)-w(\Delta)|} - 1 & \frac{w(\Delta)}{|w(0)-w(\Delta)|}\\ \frac{w(\Delta)}{|w(0)-w(\Delta)|} & \frac{w(0)}{|w(0)-w(\Delta)|} - 1\end{pmatrix}\begin{pmatrix}v(x_1)\\v(x_2)\end{pmatrix} \tag{18}$$

from which ($M = \begin{pmatrix} a & b \\ b & a \end{pmatrix}$) we can read off:

$$a = \frac{w(0)}{|w(0) - w(\Delta)|} - 1 \qquad \text{and} \qquad b = \frac{w(\Delta)}{|w(0) - w(\Delta)|} \tag{19}$$

2.4.2 Eigenpairs

Finding the eigenpairs of M in terms of a and b is a classic 2×2 eigenvalue problem. We start with $M - \lambda I = \begin{pmatrix} a - \lambda & b \\ b & a - \lambda \end{pmatrix}$. Next, we get the characteristic equation $\det(M - \lambda I) = (a - \lambda)^2 - b^2$. From here it is clear that $(a - \lambda)^2 = b^2$ and therefore $a - \lambda = \pm b$. Thus, the eigenvalues of M are $\lambda_+ = a + b$ and $\lambda_- = a - b$.

Next, we proceeded to find the eigenvectors of M $\mathbf{v}_\pm$, which are the nontrivial solutions of $\left(M - \lambda_\pm I\right) \mathbf{v}_\pm = 0$. For $\lambda_+ = a + b$ we found:

$$\begin{pmatrix} a - (a+b) & b \\ b & a - (a+b) \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \end{pmatrix} = \begin{pmatrix} -b & b \\ b & -b \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \end{pmatrix} = 0 \quad \Longrightarrow \quad v_2 = v_1$$

$$\mathbf{v}_+ = \begin{pmatrix} 1 \\ 1 \end{pmatrix}.$$

Likewise, for $\lambda_- = a - b$,

$$\begin{pmatrix} a - (a-b) & b \\ b & a - (a-b) \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \end{pmatrix} = \begin{pmatrix} b & b \\ b & b \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \end{pmatrix} = 0 \quad \Longrightarrow \quad v_2 = -v_1$$

$$\mathbf{v}_- = \begin{pmatrix} -1 \\ 1 \end{pmatrix}.$$

And so we identified the eigenpairs:

$$\lambda_+, v_+ = a + b, \begin{pmatrix} 1 \\ 1 \end{pmatrix} \tag{20}$$

$$\lambda_-, v_- = a - b, \begin{pmatrix} -1 \\ 1 \end{pmatrix} \tag{21}$$

Filling in the values for a and b:

$$\lambda_+ = a + b = \frac{w(0)}{|w(0) - w(\Delta)|} - 1 + \frac{w(\Delta)}{|w(0) - w(\Delta)|}$$

$$\lambda_+ = \frac{w(0) + w(\Delta)}{|w(0) - w(\Delta)|} - 1$$

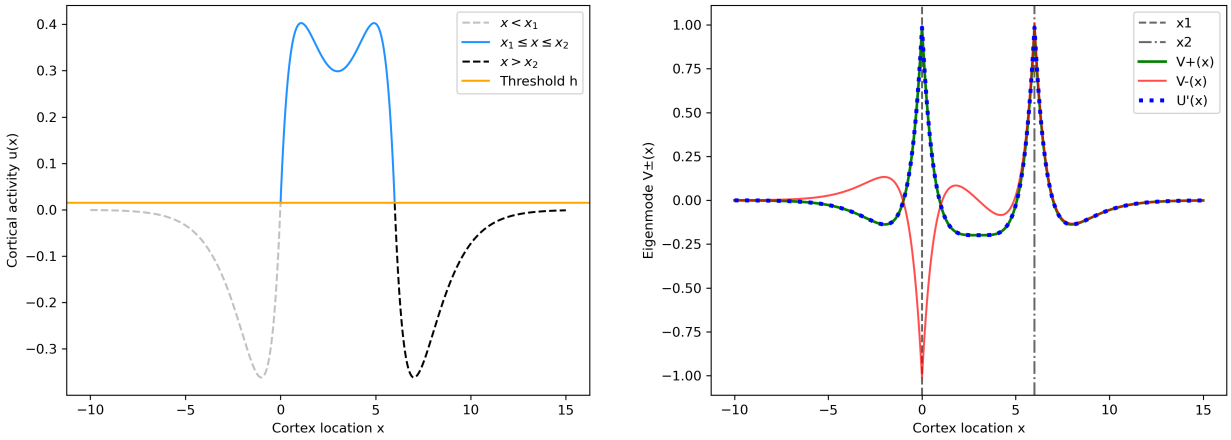
$$\lambda_- = a - b = \frac{w(0)}{|w(0) - w(\Delta)|} - 1 - \frac{w(\Delta)}{|w(0) - w(\Delta)|}$$

$$\lambda_- = \frac{w(0) - w(\Delta)}{|w(0) - w(\Delta)|} - 1 = 1 - 1 = 0$$

To obtain the eigenfunctions (eigenmodes) we use equation 16 and the obtained eigenvectors $v_+; v_-$:

$$v_-(x) = \frac{-w(x-x_1) + w(x-x_2)}{|U'(x_j)|}, \qquad C = \frac{1}{|w(0) - w(\Delta)|}$$
$$v_-(x) = C[w(x-x_2) - w(x-x_1)]$$
$$(\lambda_+ + 1)v_+(x) = \frac{w(x-x_1) + w(x-x_2)}{|U'(x_j)|}, \qquad (\lambda_+ + 1) = \frac{w(0) + w(\Delta)}{|w(0) - w(\Delta)|}$$
$$v_+(x) = \frac{w(x-x_1) + w(x-x_2)}{|w(0) - w(\Delta)|} \frac{|w(0) - w(\Delta)|}{w(0) + w(\Delta)} = \frac{w(x-x_1) + w(x-x_2)}{w(0) - w(\Delta)}$$
$$v_+(x) = C[w(x-x_1) + w(x-x_2)]$$

2.4.3 Eigenmodes $V_+(x)$ and $V_-(x)$



(a) Neural field bump

(b) Eigenmodes $v_+(x)$(green) $v_-(x)$(red)

Figure 4: Simulation of Neural field with $\Delta = 6$ and $C = 1$. $U'(x)$ in blue.

In Figure 4 we plot the a bump neural field bump $U(x)$, the corresponding eigenmodes of that bump, and $U'(x)$. We notice that while $v_+(x)$ is symmetrical and goes up at both x_1 and x_2, $v_-(x)$ is strangely asymmetrical, an goes down at x_1 and then joins $v_+(x)$ at x_2. In Figure 4b can also nicely see that $U'(x) = v_+(x)$, which was expected from the formulation of their equations (which can be made equal using the fact that $w(z)$ is even, as discussed).

2.5 Stability of $U(x)$

Ignoring the typo and following the lecture where it was emphasized that the 0 eigenvalue does not contribute to stability, we concentrate on the sign of λ_+ to determine stability. We have shown before that $w(0) = 1$:

$$\lambda_+ = \frac{w(0) + w(\Delta)}{|w(0) - w(\Delta)|} - 1 = \frac{1 + w(\Delta)}{|1 - w(\Delta)|} - 1$$
$$= \frac{1 + w(\Delta) - 1 + w(\Delta)}{|1 - w(\Delta)|}$$
$$\lambda_+ = \frac{2w(\Delta)}{|1 - w(\Delta)|} < 0 \qquad \text{if } w(\Delta) < 0$$

Since the sign of λ_+ in the final equation depends only on the numerator ($2w(\Delta)$), and λ_+ is the nonzero eigenvalue that determines stability of bumps, it is clear that a bump is stable if $w(\Delta) < 0$.

2.6 Neural field bifurcation diagram

Figure 5, shows a complete bifurcation diagram (obtained through Python simulation) with color-coded bump-width and bump-stability. It seems clear that the wide branch is stable and the narrow branch is unstable. These branches meet at $(\Delta, h) = (1, 1/e)$ and undergo a saddle-node bifurcation, after which there are no more solutions. The reason why the wide branch is stable is because as we showed earlier, $\phi(1) = 1/e$ is the maximum of $\phi(\Delta)$ and that is also where the saddle node bifurcation occurs. For $\Delta < 1$, $\phi(\Delta)$ keeps increasing towards the peak at 1 ($\phi'(\Delta) = w(\Delta) > 0$) which is hallmark of instability. Conversely, for $\Delta > 1$, $\phi(\Delta)$ only decreases ($\phi'(\Delta) = w(\Delta) < 0$), which as we showed above keeps all bumps with $\Delta > 1$ stable. Graphically, this can also be seen in the bifurcation diagram as positive slope in the unstable branch and negative slope in the stable branch.

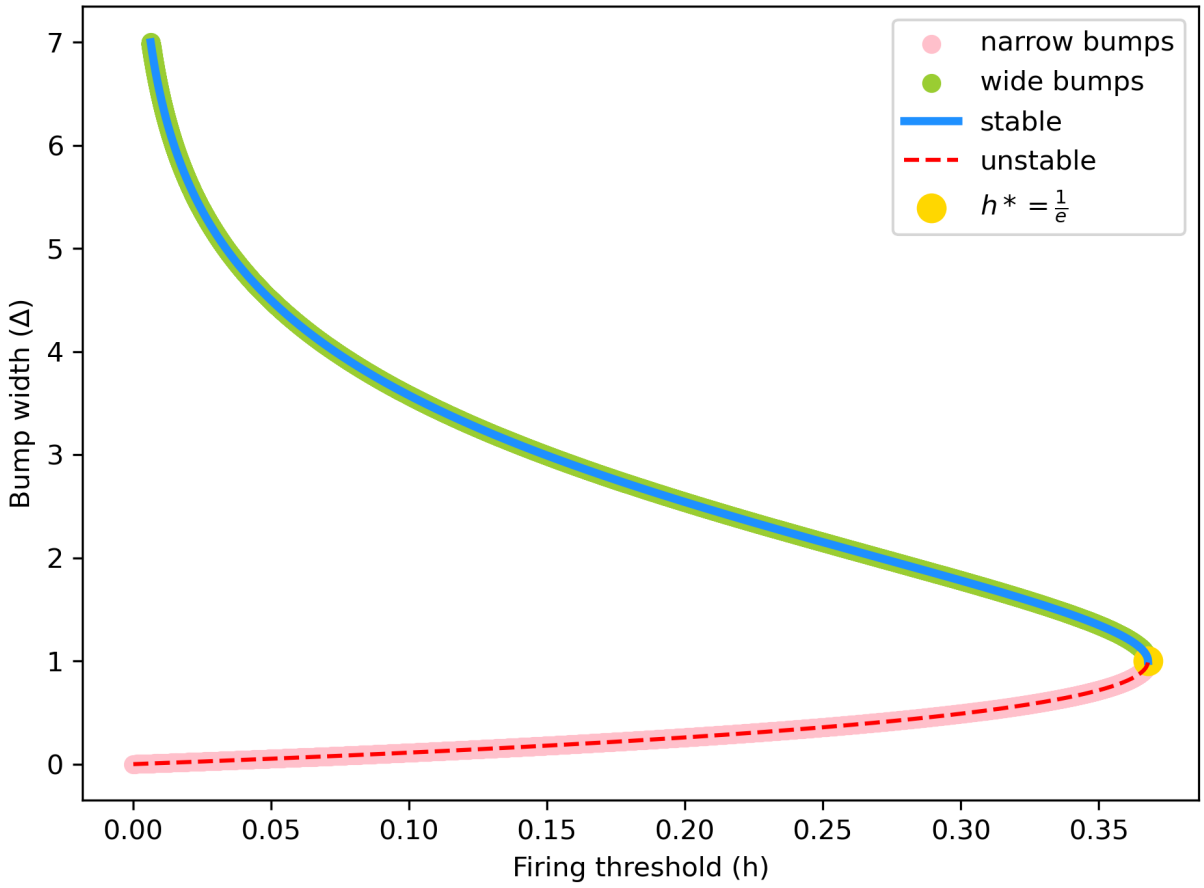


Figure 5: Complete Neural field bifurcation diagram. Wide bumps in green, narrow bumps in pink, stable bumps in blue, unstable bumps in red and saddle-node bifurcation in gold.

3 Neural Field Simulation

For this simulation, we make a few changes to our neural field model. Firstly, out model now lives on a ring Ω with width 20 ($x \in \Omega = [-10, 10)$). Second, we exchange the Heaviside firing rate function for the sigmoidal $f(u, h) = \frac{1}{1+e^{-10*(u-h)}}$. We keep the mexican hat kernel $w(z)$.

3.1 Ring neural field

To facilitate comparison of the model with the previous one, we set $\kappa = 1$. We introduce parameter α to vary the initial bump width Δ of the initial conditions:

$$u_0(x) = \frac{1}{\cosh^2(0.5\alpha x)} \tag{22}$$

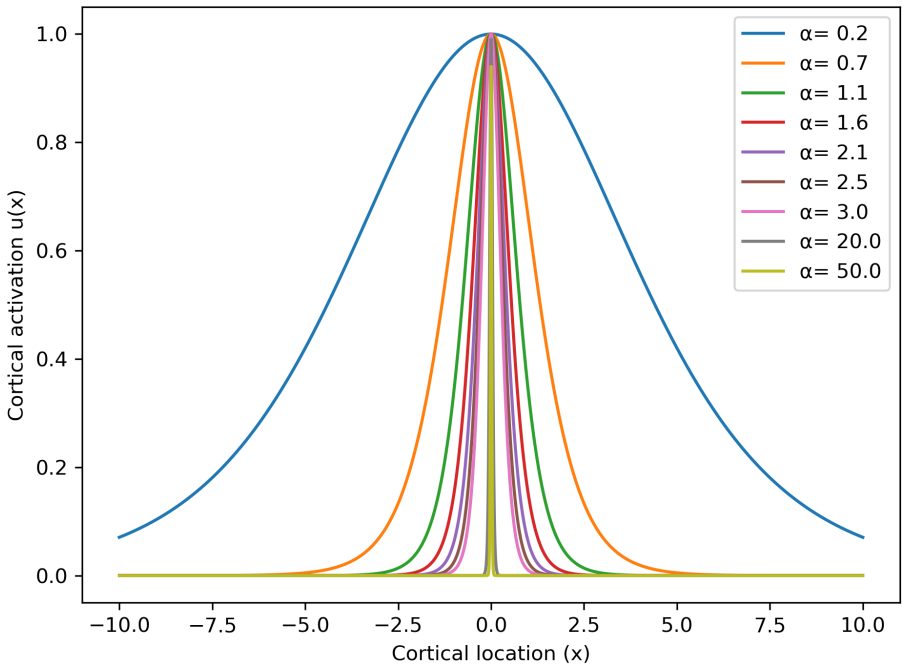


Figure 6: Initial conditions $u_0(x)$

As can be seen in Figure 6, small α gives a broad bump and large α gives a narrow localized bump. Applying these nine initial conditions to the neural field (with threshold set to $h = 0.2$) gets us the space-time activation profiles in Figure7. Very wide bumps ($\alpha < 0.4$), were very interesting and produced unexpected patterns. These initial conditions, covered and activated most of the neural ring at t_0. If threshold was low enough (or κ high enough), these initial conditions would result in multiple regularly-spaced bump peaks, all emanating from the initial broad bump peak (eg. $\alpha = 0.2$). This was behavior that could not be seen in the model investigated in question 2, and most likely this was some Turing-like behavior due to the periodic ring structure of this neural field and because these very broadly activating initial conditions put the neural field into nearly homogenous state. Fortunately, outside of this peculiar set of solutions, the rest of the findings matched our theory from question 2. Namely through a wide range of initial conditions, bumps converged to a stable bump of fixed width Δ (given that h was fixed), as would be predicted from our analyses. Relatively wide bumps (from the "wide" branch) even decreased their width as they settled into the optimal width for the given h (e.g. $\alpha = 0.67$). As initial state bump width kept decreasing, bumps started having to broaden to reach the steady state bump width on the stable branch. Eventually, we tested narrow bumps ("narrow-unstable branch") (e.g. $\alpha \in (20, 50)$. Even very narrow initial bumps (e.g.$\alpha = 20$) could quite quickly increase their width and settle into the steady state. Finally, excessively narrow initial bumps for the given threshold h (e.g.$\alpha = 50$), failed to reach the unstable narrow branch and collapsed after the initial impulse. This nicely complements our previous investigation and shows that insights from the Amari approach can be extended to ring neural fields and sigmoid firing rate functions.

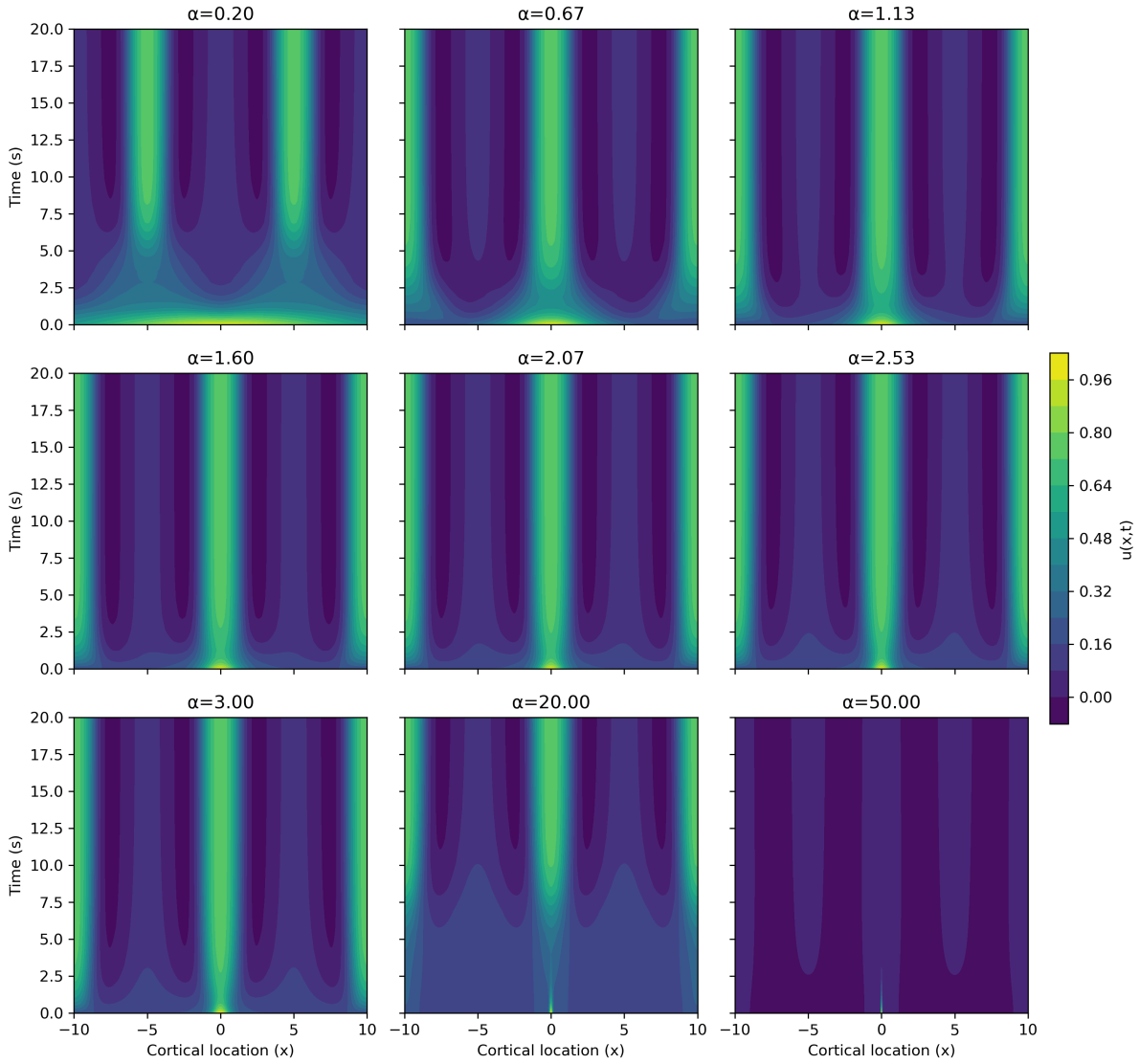


Figure 7: Bumps on ring neural field across bump widths. α is inversely proportional to bump width Δ.