A Finite Element Method for Neural Fields on Surfaces

Aaron Hagstrom
Adviser: Dr. Evan Gawlik

A thesis presented for the degree of
Master of Arts

Mathematics
University of Hawaii at Manoa
USA

1 Abstract

We introduce a finite element method for simulating neural fields on surfaces and we make observations on how neural field dynamics and the underlying synchrony might depend on the nature of the surface. The system of partial differential equations that we study is a generalization to curved surfaces of a system that has been previously used to model neural field dynamics on planar domains. We also conduct a Turing instability analysis through numerical methods to determine how patterns created by a neural field are dependent on curvature. This is a worthwhile research direction because of the need to understand neuronal behavior on the irregular curvature of the brain.

2 The Finite Element Method

We motivate the problem by beginning with solving the Poisson Equation, the partial differential equation.

$$\begin{cases} -\Delta u = f & \text{in} \quad \Omega \\ u = 0 & \text{on} \quad \partial\Omega \end{cases} \tag{1}$$

where $\Omega \subset \mathbf{R}^n$ and f is a given function.
We first define a classical or strong solution of this system as a function $u \in C^2(\Omega) \cap C^0(\overline{\Omega})$ that satisfies (1). We define

$$C^k(\Omega) = \{u : \Omega \to \mathbf{R} |$$
u and all of its partial derivatives of order $\leq k$ are continuous on $\Omega\}$

The next task is to derive the 'weak formulation'. A weak solution of the Poisson equation is a function $u \in V$ satisfying

$$\int_\Omega \nabla u \cdot \nabla v = \int_\Omega f v dx \quad \forall v \in V$$

For the 2D case,

$$V = \left\{ u \in L^2(\Omega) \middle| \frac{\partial u}{\partial x}, \frac{\partial u}{\partial y} \in L^2(\Omega), \quad u = 0 \quad \text{on} \quad \partial\Omega \right\}$$

where V is called a Sobolev space (which is a type of Hilbert space) and the derivatives are called 'weak derivatives.' The space L^2 is a Lebesgue space. That is,

$$L^2(\Omega) = \left\{ u : \Omega \to \mathbf{R} \middle| \int_\Omega u^2 dxdy < \infty \right\}$$

We derive the weak formulation by means of the divergence theorem. For ev-

ery smooth function v that vanishes on $\partial\Omega$, we have

$$\begin{aligned}\int_\Omega (fv)dxdy &= \int_\Omega (-\Delta uv)dxdy \\ &= \int_\Omega -div(\nabla u)vdxdy \\ &= \int_\Omega -div(\nabla uv)dxdy + \int_\Omega (\nabla u \cdot \nabla v)dxdy \\ &= \int_{\partial\Omega} -(\nabla uv)\cdot nds + \int_\Omega (\nabla u \cdot \nabla v)dxdy \\ &= \int_{\partial\Omega} -v\frac{\partial u}{\partial n}ds + \int_\Omega (\nabla u \cdot \nabla v)dxdy \\ &= \int_\Omega (\nabla u \cdot \nabla v)dxdy\end{aligned}$$

The third step follows from the product rule applied to $div(\nabla uv)$ The third to last step follow from Green's Theorem. The last step follows from $v = 0$ on the boundary.
In abstract terms, we want to find a $u \in V$ such that

$$a(u,v) = \ell(v) \quad \forall v \in V$$

where $a : V \times V \to \mathbf{R}$ is a bilinear form that is symmetric and positive and ℓ is a linear functional. Here,

$$\begin{aligned}a(u,v) &= \int_\Omega (\nabla u \cdot \nabla v)dxdy \\ \ell(v) &= \int_\Omega (fv)dxdy\end{aligned}$$

To solve this problem, we can define a finite dimensional subspace V_h of V, consisting of continuous, piecewise linear functions and thereby we can discretize the weak formulation in the following terms: Find a $u_h \in V_h$ such that

$$a(u_h, v_h) = \ell(v_h) \quad \forall v_h \in V_h$$

where

$$V_h = \{u_h \in C^0(\overline{\Omega})|u_h|_T \quad \text{is linear} \quad \forall T \in T_h \quad \text{and} \quad u_h = 0 \quad \text{on} \quad \partial\Omega\}$$

where T_h is a triangulation of Ω or a set of triangles $T_1, ...T_M$ such that

$$\begin{aligned}T_h = \{T_1, T_2, ..., T_m|\overline{\Omega} = \cup_{i=1}^m T_i \quad \forall i \neq j, T_i \cap T_j \\ \text{is either empty, a common vertex, or a common edge}\}\end{aligned}$$

The next task is to solve for u_h.
To do this, we first define a basis for the space V_h. Define $N = \text{Dim}(V_h)$. If $N_{\overline{\Omega}}$ is the total number of vertices and $N_{\partial\Omega}$ is the total number of vertices on $\partial\Omega$, then

$$N = N_{\overline{\Omega}} - N_{\partial\Omega}$$

Then, define the basis of V_h as $\{\phi_j(x,y)\}_{j=1}^N$, where

$$\phi_i(x_j,y_j) = \begin{cases} 1 & \text{if} \quad i=j \\ 0 & \text{if} \quad i \neq j \end{cases}$$

where (x_j, y_j) is the position of the jth vertex. Our solution will be a weighted sum of these basis functions. That is:

$$u_h(x,y) = \sum_{i=1}^{N} c_i \phi_i(x,y)$$
$$v_h(x,y) = \sum_{j=1}^{N} d_j \phi_j(x,y)$$

Substituting these functions into

$$a(u_h, v_h) = \ell(v_h)$$

and defining the "stiffness" matrix as

$$\begin{aligned} A_{ij} &= a(\phi_i, \phi_j) \\ &= \sum_{T \in T_h} \int_T (\nabla \phi_i \cdot \nabla \phi_j) dxdy \end{aligned}$$

and defining

$$\begin{aligned} b_j &= \ell(\phi_j) \\ &= \sum_{T \in T_h} \int_T f \phi_j dxdy \end{aligned}$$

we get

$$\begin{aligned} a(u_h, v_h) &= \sum_{i,j} a(c_i \phi_i, d_j \phi_j) \\ &= \sum_{i.j} c_i a(\phi_i, \phi_j) d_j \\ &= \sum_{i,j} c_i A_{ij} d_j \\ &= \mathbf{c^t A d} \end{aligned}$$

and

$$\begin{aligned} \ell(v_h) &= \sum_j \ell(\phi_j) d_j \\ &= \mathbf{b^t d} \end{aligned}$$

and so,

$$\begin{aligned} \mathbf{c^t A d} &= \mathbf{b^t d} \quad \forall \mathbf{d} \in \mathbf{R}^N \\ \iff \mathbf{c^t A} &= \mathbf{b^t} \\ \iff \mathbf{A^t c} &= \mathbf{b} \\ \iff \mathbf{A c} &= \mathbf{b} \\ \iff \mathbf{c} &= \mathbf{A^{-1} b} \end{aligned}$$

2.1 Natural Coordinates

We use what are called *natural coordinates* so that we can use Gaussian quadrature to compute the integrals appearing in the equations for a and ℓ,
We would like to map the vertices of an element $T \in T_h$ in the triangulation in the xy plane to the vertices of a reference triangle $\hat{T}$ in the $\hat{x}\hat{y}$ plane, that is, to a right triangle in the first quadrant of the Real plane having one vertex at the origin and with two sides of length 1. Define the mapping

$$\begin{aligned} F : \hat{T} &\to T \\ (\hat{x}, \hat{y}) &\mapsto (x, y) \end{aligned}$$

Denote the vertex coordinates of T as

$$\begin{aligned} &(x_{i_1}, y_{i_1}) \\ &(x_{i_2}, y_{i_2}) \\ &(x_{i_3}, y_{i_3}) \end{aligned}$$

Denote the vertex coordinates of $\hat{T}$ as

$$\begin{aligned} &(0,0) \\ &(1,0) \\ &(0,1) \end{aligned}$$

Define the map between $\hat{T}$ and T as

$$\begin{bmatrix} x \\ y \end{bmatrix} = F(\hat{x}, \hat{y}) = \begin{bmatrix} x_{i_1} \\ y_{i_1} \end{bmatrix} + \begin{bmatrix} x_{i2} - x_{i1} & x_{i3} - x_{i1} \\ y_{i2} - y_{i1} & y_{i3} - y_{i1} \end{bmatrix} \begin{bmatrix} \hat{x} \\ \hat{y} \end{bmatrix}$$

Define $\hat{\phi}_a(\hat{x}, \hat{y}) = \phi_{i_a}(x, y) = \phi_{i_a}(F(\hat{x}, \hat{y}))$ for $a = 1, 2, 3$
$\hat{\phi}_1(\hat{x}, \hat{y})$ is the unique linear function on $\hat{T}$ satisfying

$$\begin{cases} \hat{\phi}_1(0,0) = 1 \\ \hat{\phi}_1(1,0) = 0 \\ \hat{\phi}_1(0,1) = 0 \end{cases}$$

$\hat{\phi}_2(\hat{x}, \hat{y})$ is the unique linear function on $\hat{T}$ satisfying

$$\begin{cases} \hat{\phi}_2(0,0) = 0 \\ \hat{\phi}_2(1,0) = 1 \\ \hat{\phi}_2(0,1) = 0 \end{cases}$$

$\hat{\phi}_3(\hat{x}, \hat{y})$ is the unique linear function on $\hat{T}$ satisfying

$$\begin{cases} \hat{\phi}_3(0,0) = 0 \\ \hat{\phi}_3(1,0) = 0 \\ \hat{\phi}_3(0,1) = 1 \end{cases}$$

Solving these systems of equations, we get that

$$\hat{\phi}_1(\hat{x},\hat{y}) = \phi_{i_1}(x,y) = \phi_{i_1}(F(\hat{x},\hat{y})) = 1 - \hat{x} - \hat{y}$$
$$\hat{\phi}_2(\hat{x},\hat{y}) = \phi_{i_2}(x,y) = \phi_{i_2}(F(\hat{x},\hat{y})) = \hat{x}$$
$$\hat{\phi}_3(\hat{x},\hat{y}) = \phi_{i_3}(x,y) = \phi_{i_3}(F(\hat{x},\hat{y})) = \hat{y}$$

And so,

$$\hat{\nabla}\phi_1 = \begin{bmatrix} -1 \\ -1 \end{bmatrix}$$
$$\hat{\nabla}\phi_2 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$
$$\hat{\nabla}\phi_3 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

Observe that

$$\hat{\nabla}\phi_a = \begin{bmatrix} \frac{\partial \hat{\phi}_a}{\partial \hat{x}} \\ \frac{\partial \hat{\phi}_a}{\partial \hat{y}} \end{bmatrix} = \begin{bmatrix} \frac{\partial x}{\partial \hat{x}} & \frac{\partial y}{\partial \hat{x}} \\ \frac{\partial x}{\partial \hat{y}} & \frac{\partial y}{\partial \hat{y}} \end{bmatrix} \begin{bmatrix} \frac{\partial \phi_{i_a}}{\partial x} \\ \frac{\partial \phi_{i_a}}{\partial y} \end{bmatrix} = J^T \nabla \phi_{i_a}$$

So,

$$\nabla \phi_{i_a} = J^{-T} \hat{\nabla} \hat{\phi}_a$$

Also,

$$dydy = |J| d\hat{x} d\hat{y}$$

So,

$$\int_T \nabla \phi_{i_a} \cdot \nabla \phi_{i_b} dxdy = \int_{\hat{T}} (J^{-T} \hat{\nabla} \hat{\phi}_a) \cdot (J^{-T} \hat{\nabla} \hat{\phi}_b) |J| d\hat{x} d\hat{y}$$

In the same way,

$$\int_T f \phi_{i_a} dxdy = \int_{\hat{T}} \hat{f} \hat{\phi}_a |J| d\hat{x} d\hat{y}$$

where $\hat{f}(\hat{x},\hat{y}) = f(x,y) = f(F(\hat{x},\hat{y}))$.

2.2 Time Dependent Heat Equation

To motivate time-dependent neural field problems, we will examine the time-dependent heat equation, which we can formulate as the following:

$$\begin{cases} -\Delta u & = f - \frac{\partial u}{\partial t} \quad \text{in} \quad \Omega \\ u & = 0 \quad \text{on} \quad \partial\Omega \end{cases} \tag{2}$$

where t is time.

We want to find $u \in H_0^1(\Omega)$ such that

$$\int_\Omega \frac{\partial u}{\partial t} v + \int \nabla u \cdot \nabla v = \int f v$$

$\forall v \in H_0^1(\Omega)$. We can reframe this as finding a $u_h \in V_h$ as in the case of the Poisson problem such that

$$\int_\Omega \frac{\partial u_h}{\partial t} v_h + \int_\Omega \nabla u_h \cdot \nabla v_h = \int_\Omega f v_h$$

$\forall v_h \in V_h$
Next, choosing $\{\phi_i\}_{i=1}^N$ as a basis for V_h, we can write that

$$u_h(t,x,y) = \sum_{i=1}^N c_i(t)\phi_i(x,y)$$
$$v_h(t,x,y) = \sum_{j=1}^N d_j(t)\phi_j(x,y)$$

This leads to

$$\sum_{i,j}^N \int_\Omega \frac{dc_i}{dt}\phi_i(x,y)\cdot d_j\phi_j(x,y) + \int_\Omega c_i(t)\nabla\phi_i(x,y)\cdot d_j\nabla\phi_j(x,y) = \sum_j \int f d_j \phi_j(x,y)$$
$$\sum_{i,j} \frac{dc_i}{dt}\left(\int \phi_i \cdot \phi_j\right)d_j + c_i\left(\int_\Omega \nabla\phi_i \cdot \nabla\phi_j\right)d_j = \sum_j d_j \int_\Omega f\phi_j$$

Replacing $\int_\Omega \phi_i \cdot \phi_j$ with M_{ij} and replacing $\int_\Omega \nabla\phi_i \cdot \nabla\phi_j$ with A_{ij} and $\int_\Omega f\phi_j$ with b_j, we get

$$\sum_{i,j} \frac{dc_i}{dt} M_{ij} d_j + c_i A_{ij} d_j = \sum_j d_j b_j$$

Or as matrices, we can write as

$$\dot{\mathbf{c}}^\mathbf{T}\mathbf{M}\mathbf{d} + \mathbf{c}^\mathbf{T}\mathbf{A}\mathbf{d} = \mathbf{b}^\mathbf{T}\mathbf{d}$$

$\forall \mathbf{d}$. So,

$$\dot{\mathbf{c}}^\mathbf{T}\mathbf{M} + \mathbf{c}^\mathbf{T}\mathbf{A} = \mathbf{b}^\mathbf{T}$$
$$\mathbf{M}^\mathbf{T}\dot{\mathbf{c}} + \mathbf{A}^\mathbf{T}\mathbf{c} = \mathbf{b}$$
$$\mathbf{M}\dot{\mathbf{c}} + \mathbf{A}\mathbf{c} = \mathbf{b}$$

This is called the *semi-discrete Galerkin* formulation because we have only discretized the spatial variable but not the time derivative. One of the more common methods to discretize the time derivative is the *θ-method*, which involves replacing the time derivative with the difference

$$\frac{\partial \mathbf{c}}{\partial t} \approx \frac{\mathbf{c^{n+1}} - \mathbf{c^n}}{\Delta t} \tag{3}$$

where Δt is the time increment, and $t_{n+1} = t_n + \Delta t$. We introduce a *relaxation parameter* θ and write $\mathbf{c}$ as

$$\mathbf{c} = \theta \mathbf{c^{n+1}} + (1-\theta)\mathbf{c^n} \tag{4}$$

for $t_n \leq t \leq t_{n+1}$ where θ is usually specified as being in the range $0 \leq \theta \leq 1$. This parameter controls the accuracy and stability of the algorithm. Setting $\theta = 0$, we get $\mathbf{c} = \mathbf{c^n}$

$$\begin{aligned} \mathbf{M}\left(\frac{\mathbf{c^{n+1}} - \mathbf{c^n}}{\Delta t}\right) + \mathbf{A}\mathbf{c^n} &= \mathbf{b} \\ \mathbf{M}\mathbf{c^{n+1}} &= \mathbf{M}\mathbf{c^n} + \Delta t(\mathbf{b} - \mathbf{A}\mathbf{c^n}) \\ \mathbf{c^{n+1}} &= \mathbf{M^{-1}}(\mathbf{M}\mathbf{c^n} + \Delta t(\mathbf{b} - \mathbf{A}\mathbf{c^n})) \end{aligned}$$

which is the solution.

2.3 FEM Implementation

We take six general steps when implementing the finite element method in MATLAB.
(1) Input data for boundary conditions, forces, and geometry of domain.
(2) Define the triangulation T_h
(3) Compute the stiffness matrix, mass, and force matrices.
(4) Assemble the global stiffness matrix $\mathbf{A}$ and vector $\mathbf{b}$.
(5) Solve the system $\mathbf{Ac} = \mathbf{b}$
(6) Plot the resulting solution.

3 Error Analysis

3.1 Energy Norm (i.e. a-norm)

We would like to understand how much error results from an FEM formulation. Ultimately, we are interested in analyzing finite element methods for neural fields on surfaces, but for now, we will be content with a 1D formulation that gives the general idea of what we are trying to accomplish.
Let $V_h \subset V$ be a finite-dimensional subspace of V, where V is a Hilbert Space, and let $a : V \times V \to \mathbf{R}$ be a symmetric bilinear form satisfying three conditions. The first and last are continuity properties and the second is called coercivity.

$$\begin{cases} |a(u,v)| & \leq M\|u\|_V\|v\|_V \quad \forall u, v \in V \\ a(u,u) & \geq \alpha\|u\|_V^2 \quad \forall u \in V \\ \|\ell\|_{V'} & = \sup\limits_{v \in V} \frac{|\ell(v)|}{\|v\|_V} < \infty \end{cases}$$

where M and α are positive constants.
We can first conduct an error analysis of the a-norm of the error, where

$$\|u\|_a = a(u,u)^{\frac{1}{2}}$$

We can prove that if $u \in V$ satisfies

$$a(u, v) = \ell(v) \quad \forall v \in V$$

and $u_h \in V_h$ satisfies

$$a(u_h, v_h) = \ell(v_h) \quad \forall v_h \in V_h$$

then,

$$\|(u - u_h)\|_a = \inf_{v_h \in V_h} \|u - v_h\|_a$$
$$\|u\|_a = a(u, u)^{1/2}$$

The weak formulation implies that

$$a(u, v_h) = \ell(v_h) \quad \forall v_h \in V_h$$

and the Galerkin formulation implies that

$$a(u_h, v_h) = \ell(v_h) \quad \forall v_h \in V_h$$

Subtracting the second expression from the first, we obtain

$$a(u - u_h, v_h) = 0 \quad \forall v_h \in V_h$$

which can be interpreted as "$u - u_h$ is a-orthogonal to V_h."
This fact becomes useful in evaluating the expression

$$\begin{aligned} \|u - u_h\|_a^2 &= a(u - u_h, u - u_h) \\ &= a(u - u_h, u - v_h) + a(u - u_h, v_h - u_h) \\ &= a(u - u_h, u - v_h) \\ &\leq \|u - u_h\|_a \|u - v_h\|_a \\ \forall v_h \in V_h & \end{aligned}$$

So, we have

$$\|u - u_h\|_a^2 \leq \|u - u_h\|_a \|u - v_h\|_a \quad \forall v_h \in V_h$$

So,

$$\|u - u_h\|_a = \inf_{v_h \in V_h} \|u - v_h\|_a$$

3.2 Discretization Error

Now we will analyze the error in the V-norm.
We can prove that

$$\|u - u_h\|_V \leq \sqrt{\frac{M}{\alpha}} \inf_{v_h \in V_h} \|u - v_h\|_V$$

by the following argument

$$\begin{aligned} \alpha\|u - u_h\|_V^2 &\leq a(u - u_h, u - u_h) \\ &= \|u - u_h\|_a^2 \\ &\leq \|u - v_h\|_a^2 \\ &= a(u - v_h, u - v_h) \\ &\leq M\|u - v_h\|_V^2 \\ \forall v_h \in V_h & \end{aligned}$$

So,

$$\begin{aligned} \|u - u_h\|_V \leq \sqrt{\frac{M}{\alpha}}\|u - v_h\|_V \\ \forall v_h \in V_h \end{aligned}$$

3.3 Interpolation Error

We wish to consider a one-dimensional problem $-u'' = f$ on $\Omega = (0,1)$ with boundary conditions $u(0) = u(1) = 0$, and we're partitioning the interval $(0,1)$ into subintervals of length h and assume $u \in H^2(\Omega)$. Let

$$\begin{cases} V & = H_0^1(\Omega) \\ V_h & = \{v_h \in C^0(\Omega) | v_h|_{jh,(j+1)h} \text{ is linear } \forall j \text{ and } v_h(0) = v_h(1) = 0\} \\ \|u\|_a & = (\int_\Omega u'(x)^2 dx)^{\frac{1}{2}} = |u|_{H^1(\Omega)} \\ \|u\|_V & = \|u\|_{H^1} = (\int_0^1 u(x)^2 dx + \int_0^1 u'(x)^2 dx)^{\frac{1}{2}} = (\|u\|_{L^2(\Omega)}^2 + |u|_{H^1(\Omega)}^2)^{\frac{1}{2}} \\ a(u,v) & = \int_\Omega u'(x)v'(x)dx \\ \ell(v) & = \int_\Omega f(x)v(x)dx \end{cases}$$

The general theory above showed that

$$|u - u_h|_{H^1(\Omega)} = \inf_{v_h \in V_h} |u - v_h|_{H^1(\Omega)}$$

We want to show that the right-hand side of this equality is small.
We need to find some $v_h \in V_h$, (where v_h is an arbitrary piecewise linear function) such that $|u - v_h|_{H^1(\Omega)}$ is small. So, we need a specific $v_h \in V_h$ such that this difference is small. We can choose v_h to be a piecewise linear interpolant of u. In other words $v_h = I_h u$, meaning that u is approximated with an interpolant. We can express v_h as a summation over basis functions defined at gridpoints x_i.

As noted above, we are defining v_h as interpolant of u and so

$$\begin{aligned} v_h(x) &= I_h u(x) \\ &= \sum_{i=1}^{N} u(x_i)\phi_i(x) \end{aligned}$$

We know that

$$\begin{aligned} v_h(x_j) &= \sum_{i=1}^{N} u(x_i)\phi_i(x_j) \\ &= u(x_j) \end{aligned}$$

We can prove that in one-dimension

$$|u - v_h|_{H^1(\Omega)} \leq h|u|_{H^2(\Omega)}$$

Denote

$$e(x) = u(x) - v_h(x)$$

such that

$$\begin{aligned} |u - v_h|^2_{H^1(\Omega)} &= \int_0^1 e'(x)^2 dx \\ &= \sum_{j=0}^{N} \int_{jh}^{(j+1)h} e'(x)^2 dx \end{aligned}$$

By the Mean Value Theorem, there exists $\xi \in [jh, (j+1)h]$ such that

$$e'(\xi) = \frac{e((j+1)h) - e(jh)}{h} = 0$$

And so, $\forall x \in [jh, (j+1)h]$,

$$\begin{aligned} e'(x) &= \int_{\xi}^{x} e''(y)dy \\ |e'(x)| &= \left| \int_{\xi}^{x} e''(y)dy) \right| \\ &\leq |x - \xi|^{\frac{1}{2}} \left(\int_{\xi}^{x} e''(y)^2 dy \right)^{\frac{1}{2}} \\ &\leq h^{\frac{1}{2}} \left(\int_{jh}^{(j+1)h} e''(y)^2 dy \right)^{\frac{1}{2}} \\ \int_{jh}^{(j+1)h} |e'(x)|^2 dx &\leq \int_{jh}^{(j+1)h} h \int_{jh}^{(j+1)h} e''(y)^2 dy dx \\ &= h^2 \int_{jh}^{(j+1)h} e''(y)^2 dy \\ &= h^2 \int_{jh}^{(j+h)j} u''(y)^2 dy \end{aligned}$$

Since,

$$e'' = u'' - v_h'' = u'' \text{ on } (jh, (j+1)h).$$

So,

$$\sum_{j=0}^{N} \int_{jh}^{(j+1)h} |e'(x)|^2 dx \leq h^2 \sum_{j=0}^{N} \int_{jh}^{(j+1)h} u''(y)^2 dy$$

$$|e|^2_{H^1(\Omega)} \leq h^2 |u|^2_{H^2(\Omega)}$$

$$|u - v_h|_{H^1(\Omega)} \leq h|u|_{H^2(\Omega)}$$

This means that we can reduce the error between u and v_h in the H^1 seminorm significantly just by making h arbitrarily small.
To summarize, if $u \in V$ is the solution of $a(u, v) = \ell(v) \quad \forall v \in V$ and $u_h \in V_h$ is the solution of $a(u_h, v_h) = \ell(v_h) \quad \forall v_h \in V_h$, then

$$|u - u_h|_{H^1(\Omega)} \leq h|u|_{H^2(\Omega)}$$

if $u \in H^2(\Omega)$. This means that the derivatives of u_h are converging to the derivatives of u at a linear rate.

4 Surface PDEs

4.1 The Basics

We are interested in simulating neural fields, reformulated as partial differential equations, on surfaces. These types of equations currently have applications in image analysis, cellular biology, fluid dynamics, and materials sciences. Begin by defining a smooth, connected, oriented hypersurface in $\mathbf{R}^{n+1}$, $n = 1, 2$ [8].
Let $d = d(x)$ with $x \in \mathbf{R}^{n+1}$ be a smooth level set function. Let N be an open subset of $\mathbf{R}^{n+1}$, where $\nabla d \neq 0$.
Define a surface Γ as

$$\Gamma = \{x \in N | d(x) = 0\}$$

Assume d is twice differentiable,

$$d \in C^2(N)$$

Fix the orientation of Γ such that the normal $\hat{n}$ to Γ is in the direction of increasing d.
The normal vector field can be defined as

$$\hat{n}(x) = \frac{\nabla d(x)}{|\nabla d(x)|}.$$

where $x \in \mathbf{R}^{n+1}$.

Denote the projection at x onto the tangent space of Γ as having the i,j-element

$$P(x)_{ij} = \delta_{ij} - \hat{n}(x)_i \hat{n}(x)_j$$

We can choose N and a $d(x)$ (i.e. one possible choice is a signed distance function) such that for every $x \in N$ there exists a unique $a(x) \in \Gamma$ such that

$$x = a(x) + d(x)\hat{n}(a(x))$$

Define the tangential gradient on Γ by

$$\nabla_\Gamma \eta = \nabla\eta - \nabla\eta \cdot \hat{n}\hat{n} = P\nabla\eta$$

for a function η defined on N, where $\nabla\eta$ denotes the gradient on $\mathbf{R}^{n+1}$
We know that

$$\nabla_\Gamma \eta \cdot \hat{n} = 0.$$

Let $\underline{D}$ be the tangential gradient. The components of the tangential gradient are defined as

$$\nabla_\Gamma \eta = (\underline{D}_1\eta, ..., \underline{D}_{n+1}\eta)$$

Define the Laplace-Beltrami operator on Γ as the tangential divergence of the tangential gradient:

$$\Delta_\Gamma \eta = \nabla_\Gamma \cdot \nabla_\Gamma \eta = \sum_{i=1}^{n+1} \underline{D}_i \underline{D}_i \eta$$

If we denote the outer unit normal vector of $\partial\Gamma$ by μ we can see that

$$\int_\Gamma \underline{D}_i \eta = -\int_\Gamma \eta H \hat{n}_i + \int_{\partial\Gamma} \eta \mu_i$$

where $H = -\nabla \cdot \hat{n}$ is the mean curvature (the sum of the principal curvatures) with respect to $\hat{n}$.
This equation yields the divergence theorem for

$$\xi = (\xi_1, \xi_2, ..., \xi_{n+1}) :$$

$$\int_{\partial\Gamma} \xi \cdot \mu = \int_\Gamma \nabla_\Gamma \cdot \xi + \int_\Gamma \xi \cdot \hat{n} H$$

Green's formula on the surface Γ is

$$\int_\Gamma \nabla_\Gamma \xi \cdot \nabla_\Gamma \eta = \int_{\partial\Gamma} \xi \nabla_\Gamma \eta \cdot \mu - \int_\Gamma \xi \Delta_\Gamma \eta$$

Much of our results are computed on the sphere. In this case, we let

$$\Gamma = \{x \in \mathbf{R}^{n+1} | |x - x_0| = R\}$$

and

$$d(x) = R - |x - x_0|$$

where R is the radius and x_0 is the center of the sphere. The normal vectors are pointing inward and mean curvature is

$$H = \frac{n}{R}$$

For Hilbert spaces on surfaces, define the following.

$$H^1(\Gamma) = \{\eta \in L^2(\Gamma) | \nabla_\Gamma \eta \in L^2(\Gamma)^{n+1}\}$$

$$H_0^1(\Gamma) = \{\eta \in H^1(\Gamma) | \eta = 0 \quad \text{on} \quad \partial\Gamma\}$$

4.2 The Jacobian and its Pseudo-determinant

In our finite element formulation on a sphere, we will need to use a different Jacobian and determinant. The Jacobian for a two-dimensional manifold immersed in $\mathbf{R}^3$ is given by

$$\begin{aligned} \mathbf{J} &= [\mathbf{J_1}, \mathbf{J_2}] \\ &= \begin{bmatrix} \frac{\partial x}{\partial \hat{x}} & \frac{\partial x}{\partial \hat{y}} \\ \frac{\partial y}{\partial \hat{x}} & \frac{\partial y}{\partial \hat{y}} \\ \frac{\partial z}{\partial \hat{x}} & \frac{\partial z}{\partial \hat{y}} \end{bmatrix} \end{aligned}$$

The Jacobian pseudo-determinant is the transformation of the volume of the differential integral measure.
The determinant is the volume of the parallelogram spanned by the two columns of $\mathbf{J}$.

$$|\mathbf{J}| = |\mathbf{J}_1 \times \mathbf{J}_2|_2.$$

where $|\cdot|_2$ is the Euclidean norm.

5 Mathematical Models of Neurons

5.1 Properties of Neurons

A neuron is an electrically excitable cell composed of a cell body (i.e. soma, where DNA is located), dendrites (tree like appendages), and one axon (which outputs signals). Neurons mainly communicate with one another by sending signals along their axons. There are about 10^{11} neurons in the human brain [6]. A neuron receives inputs from about $10,000$ other neurons through contacts on the dendritic tree called synapses, which produce changes in the membrane potential of the neuron called post-synaptic potentials (PSPs) and lead to an action potential or spike [6].
What causes neurons to spike is incoming pulses from other neurons that lead to post-synaptic potentials that are greater than the "firing threshold." Gap junctions directly connect neurons through channels [6].

5.2 Network Models

The neural circuits of the brain are extensively interconnected. One neuron in a mammalian neocortex, which comprises 90 percent of the thin outer surface of the cortex, can receive thousands of synaptic inputs. These networks play a big role in various brain tasks including short-term and associative memory [6].

We can construct network models, often called Firing Rate Models (FREs), at the microscopic level by connecting individual neurons or at the macroscopic level by coupling neuron-like units with outputs consisting of firing rates rather than action potentials. However, the main challenge is that FREs usually are not exact models of the underlying microscopic dynamics and assume the spiking activity of the neurons is asynchronous and so uncorrelated. Thus, they can't characterize the underlying system very well by themselves [6].

So, an overarching problem in mathematical and computational neuroscience is then the means by which to link the microscopic and macroscopic network descriptions of neurons in order to create a model that incorporates properties of both and that can be analyzed fairly easily and without great computational expense. For this reason, *mean-field reduction methods* [1] have been instrumental in understanding the underlying dynamics of neuronal systems including the level of synchrony within a population of neurons. With these methods, we can sometimes replace a large network of interacting neurons (i.e. oscillators) by a mean-field that exactly describes the properties of these underlying oscillators. These are called *next-generation models*. After the reduction process is complete, we obtain a low-dimensional dynamical system also known as a *neural mass model*, which is much less intractable.

5.3 Synchrony

Synchrony, when nodes oscillate in unison, is one of the most studied behaviour of coupled oscillators and can yield exotic patterns. This phenomenon can either be global or local and is thought to be integral to certain brain functions and neurological disorders such as epilepsy. Local synchrony, also called *within-population synchrony* [4] which is what we are most interested in, can be seen as oscillations within bumps, even as the surrounding activity stays constant with a low firing rate. This type of synchrony can explain a phenomenon of temporal variation in EEG/MEG frequency bands called *beta-decrease* and *beta-rebound*. That is, while an EEG scalp electrode can only detect an electric field if all individual cells act coherently – meaning they are in near global synchrony – in practise, there are fluctuations in the power of frequency bands, seeming to imply that neurons' level of synchronization is changing locally. An oscillation within a bump usually occurs when there is a collision between a Turing bifurcation and a Hopf bifurcation. Data on local synchrony can be merged with connectome data to give a new perspective on whole brain dynamics.

6 Neural Field Models

Neural fields are continuum models for analyzing large scale population dynamics of neurons that we can derive from the neural mass models. Our motivation for using a neural field is to solve the model on a surface or manifold, in this case, a sphere, that simulates the brain cortex so we can observe pattern formation and the evolution of synchrony. Visser [21] and Nunez have sought to describe generation of EEG signals in terms of a neural field (PIDE) solved on a sphere. Coombes reports exotic pattern formation, including a large rotating wave reported in EEG studies of schizophrenic patients. Coombes utilized an unorthodox combination of linear stability analysis, symmetric bifurcation theory, center manifold reduction, and numerical simulations.
Martin, in her PhD thesis [15], uses collocation methods to solve the simplest neural field model called the Amari equation on a periodic square domain, a torus, and on neuroimaging representations of a human and rat brain.
On a representation of the human brain, Martin made two observations on the connection between manifold shape and the neural field: traveling bump solutions get trapped in regions of large negative curvature when moving along non-geodesic paths (similar in case of rat brain), and when following winding paths, bump solutions often break up into two bumps and traverse the sulcus (i.e. grooves in brain) in opposite directions.
Jirsa has modeled a neural field as a nonlinear PDE and then mapped the field onto the folded cortex to help explain EEG and MEG patterns [12].
While all these researchers mentioned above, except Jirsa, studied very simple PIDEs on a surface, in this thesis, we are more interested in studying the PIDE reformulated as a PDE.

6.1 Coombes' Model

Coombes modifies a reduction model of Montbrio [4,16] by adding gap junctions which are known to promote synchrony and expanding it to a PDE model on a planar surface. He derives FREs that correspond exactly to an all-to-all coupled population of what are called all-to-all coupled Quadratic Integrate-and-Fire (QIF) neurons. This system is the neural mass model

$$\begin{cases} \tau \dot{R} = -\kappa_V R + 2RV + \frac{\gamma}{\pi\tau} \\ \tau \dot{V} = \eta_0 + V^2 - \pi^2\tau^2 R^2 + \kappa_s U \\ QU = R \end{cases} \tag{5}$$

where R is the instantaneous mean firing rate or fraction of neurons firing at a time t, V is the average membrane potential and U is the synaptic activity being driven by the mean firing rate. As for parameters, κ_s is the synaptic coupling constant, κ_V is the gap-junction coupling constant, η_0 is a constant that describes the excitability of neurons in the system, τ is some constant, γ is the degree of heterogeneity in the network, and Q is the differential operator

$$Q = \left(1 + \frac{1}{\alpha}\frac{d}{dt}\right)^2$$

where $\frac{1}{\alpha}$ is the time-to-peak of the synapse.

6.2 PDE Methods in 2D

There are two methods of translating neural fields, which are Partial Integro-Differential Equations (PIDEs), into PDEs.The first method uses the fact that the Fourier transform of the convolution of two functions is the product of their Fourier transforms.
A second method relies on fact that solutions to inhomogeneous linear differential equations can be written as the convolution of a Green's function and boundary conditions [14]. The PIDE we are interested in converting to a PDE is created in the following manner. We replace temporal derivatives with partial derivatives in the neural mass model and also replace the temporal dynamics $QU = R$ with the spatial dynamics $QU = \Psi$ where Ψ is the spatial drive [4]. Additionally,

$$\Psi = w \otimes R$$

where $\otimes$ denotes spatial interactions within the neural field model and w denotes structural connectivity (i.e. it is a function of the distance between two points). In the plane, where

$$(R, V, U) = (R(\mathbf{r}, t), V(\mathbf{r}, t), U(\mathbf{r}, t))$$

where $\mathbf{r} \in \mathbf{R}^2$ and $t \geq 0$, we define the convolution

$$[w \otimes R](\mathbf{r}, t) = \int_{\mathbf{R}^2} w(|\mathbf{r} - \mathbf{r}'|) R\left(\mathbf{r}', t - \frac{|\mathbf{r} - \mathbf{r}'|}{\nu}\right) d\mathbf{r}' \tag{6}$$

where ν denotes the speed of an action potential. Equation (6) can be rewritten as the convolution

$$\Psi(\mathbf{r}, t) = \int_{\mathbf{R}} dt' \int_{\mathbf{R}^2} d\mathbf{r}' G(\mathbf{r} - \mathbf{r}', t - t') R(\mathbf{r}', t')$$

where

$$G(\mathbf{r}, t) = w(r)\delta\left(t - \frac{r}{\nu}\right)$$

and $r = |\mathbf{r}|$. For a certain choice of w, namely

$$w(r) = \frac{(\frac{r}{2} - 1)e^{-r}}{2\pi r}$$

we can obtain a PDE model referred to as a 2D *brain-wave equation*

$$\left[\left(1 + \frac{1}{\nu}\frac{\partial}{\partial t}\right)^2 - \frac{3}{2}\nabla^2\right]^2 \Psi = -\left[\frac{1}{\nu}\frac{\partial}{\partial t}\left(1 + \frac{1}{\nu}\frac{\partial}{\partial t}\right) - \frac{3}{2}\nabla^2\right] R \tag{7}$$

Coupling this brain-wave equation with our neural mass model yields the PDE system:

$$\begin{cases} \tau\frac{\partial R}{\partial t} & = -\kappa_V R + 2RV + \frac{\gamma}{\pi\tau} \\ \tau\frac{\partial V}{\partial t} & = \eta_0 + V^2 - \pi^2\tau^2 R^2 + \kappa_s U \\ QU & = \Psi \\ \left[\left(1 + \frac{1}{\nu}\frac{\partial}{\partial t}\right)^2 - \frac{3}{2}\nabla^2\right]^2 \Psi & = -\left[\frac{1}{\nu}\frac{\partial}{\partial t}\left(1 + \frac{1}{\nu}\frac{\partial}{\partial t}\right) - \frac{3}{2}\nabla^2\right] R \end{cases} \tag{8}$$

6.3 FEM Discretization

We are considering the generalization of (8) to surfaces, which is obtained by replacing the Laplace operator Δ (or ∇^2) by the surface Laplacian Δ_Γ in (7). We can discretize the 2D neural field model (8) with a mixed finite element method using substitutions of Byrne [4]. In the first step, we break up (7), which corresponds to the last equation of (8) into six PDEs with the substitutions A_1, A_2, A_3, A_4, p, g defined as:

$$\begin{aligned}
A_1 &= 1 + \frac{1}{\nu}\frac{\partial}{\partial t}\Psi \\
A_2 &= \left(1 + \frac{1}{\nu}\frac{\partial}{\partial t}\right)A_1 - \frac{3}{2}\Delta\Psi \\
A_3 &= \left(1 + \frac{1}{\nu}\frac{\partial}{\partial t}\right)A_2 \\
A_4 &= \frac{3}{2}\Delta A_2 - \left(\frac{1}{\nu}\frac{\partial R}{\partial t} + \frac{1}{\nu^2}\frac{\partial^2 R}{\partial t^2} - \frac{3}{2}\Delta R\right) \\
p &= \left(1 + \frac{1}{\alpha}\frac{\partial}{\partial t}\right)U \\
g &= U
\end{aligned}$$

These substitutions yield the following system of six partial differential equations.

$$\begin{cases}
\frac{\partial \Psi}{\partial t} &= \nu(-\Psi + A_1) \\
\frac{\partial A_1}{\partial t} &= \nu(-A_1 + A_2 + \frac{3}{2}\Delta\Psi) \\
\frac{\partial A_2}{\partial t} &= \nu(A_3 - A_2) \\
\frac{\partial A_3}{\partial t} &= \nu(A_4 - A_3) \\
\frac{\partial p}{\partial t} &= \alpha(\Psi - p) \\
\frac{\partial g}{\partial t} &= \alpha(p - g).
\end{cases}$$

Coupling this system with the two PDEs for $\dot{R}$ and $\dot{V}$, we get the eight-equation system

$$\begin{cases}
\frac{\partial R}{\partial t} &= \frac{1}{\tau}\big(-\kappa_V R + 2RV + \frac{\gamma}{\tau\pi}\big) \\
\frac{\partial V}{\partial t} &= \frac{1}{\tau}\big(\kappa_s g - (\pi\tau R)^2 + V^2 + \eta_0\big) \\
\frac{\partial \Psi}{\partial t} &= \nu(-\Psi + A_1) \\
\frac{\partial A_1}{\partial t} &= \nu(-A_1 + A_2 + \frac{3}{2}\Delta\Psi) \\
\frac{\partial A_2}{\partial t} &= \nu(A_3 - A_2) \\
\frac{\partial A_3}{\partial t} &= \nu(A_4 - A_3) \\
\frac{\partial p}{\partial t} &= \alpha(\Psi - p) \\
\frac{\partial g}{\partial t} &= \alpha(p - g).
\end{cases} \tag{9}$$

As in the case of the time-dependent heat equation, we would like to find

$$R, V, \Psi, A_1, A_2, A_3, p, g \in V_h$$

where V_h is the space of continuous piecewise linear functions, such that our weak formulation holds for all test functions $D_h \in V_h$.

A finite element solution of the system is a collection of functions $R_h, V_h, \Psi_h, (A_1)_h, (A_2)_h, (A_3)_h, p_h, g_h \in V_h$ satisfying

$$\begin{cases} (\dot{R}_h, D_h) & = \frac{1}{\tau}(-\kappa_V(R_h, D_h) + 2(R_h V_h, D_h) + \frac{\gamma}{\pi\tau}(1, D_h)) \\ (\dot{V}_h, D_h) & = \frac{1}{\tau}(\kappa_S(g_h, D_h) - (\pi^2\tau^2(R_h^2, D_h)) + (V_h^2, D_h) + \eta_0(1, D_h)) \\ (\dot{\Psi}_h, D_h) & = \nu(-(\psi_h, D_h) + ((A_1)_h, D_h)) \\ ((\dot{A}_1)_h, D_h) & = \nu(-((A_1)_h, D_h) + ((A_2)_h, D_h) - \frac{3}{2}(\nabla\Psi_h, \nabla D_h)) \\ ((\dot{A}_2)_h, D_h) & = \nu(-((A_2)_h, D_h) + (A_3, D_h)) \\ ((\dot{A}_3)_h, D_h) & = \nu(-((A_3)_h, D_h) + ((A_4)_h, D_h)) \\ (\dot{p}_h, D_h) & = \alpha(-(p_h, D_h) + (\Psi_h, D_h)) \\ (\dot{g}_h, D_h) & = \alpha(-(g_h, D_h) + (p_h, D_h)) \end{cases}$$

for every $D_h \in V_h$, where

$$((A_4)_h, D_h) = -\frac{3}{2}(\nabla(A_2)_h, \nabla D_h) - \frac{3}{2}(\nabla R_h, \nabla D_h) - \frac{1}{\nu^2}(\ddot{R}_h, D_h) - \frac{1}{\nu}(\dot{R}_h, D_h)$$

and where

$$(\ddot{R}_h, D_h) = \frac{1}{\tau}(-\kappa_V(\dot{R}_h, D_h) + 2(V_h\dot{R}_h, D_h) + 2(R_h\dot{V}_h, D_h))$$

Choosing $\{\phi_r(x, y, z)\}_{r=1}^N$ as a basis for V_h, we can write

$$\begin{cases} R_h(x, y, z, t) & = \sum_i R_i(t)\phi_i(x, y, z) \\ V_h(x, y, z, t) & = \sum_j V_j(t)\phi_j(x, y, z) \\ \Psi_h(x, y, z, t) & = \sum_k \Psi_k(t)\phi_k(x, y, z) \\ U_h(x, y, z, t) & = \sum_l U_l(t)\phi_l(x, y, z) \\ (A_1)_h(x, y, z, t) & = \sum_m (A_1)_m(t)\phi_m(x, y, z) \\ (A_2)_h(x, y, z, t) & = \sum_n (A_2)_n(t)\phi_n(x, y, z) \\ (A_3)_h(x, y, z, t) & = \sum_o (A_3)_o(t)\phi_o(x, y, z) \\ p_h(x, y, z, t) & = \sum_s p_s(t)\phi_s(x, y, z) \\ g_h(x, y, z, t) & = \sum_q g_q(t)\phi_q(x, y, z) \end{cases} \tag{10}$$

Now, define the summations below

$$\begin{cases}
(\dot{R}_h, D_h) = \sum_{i,r}(\int \dot{R}_i(\phi_i \cdot \phi_r))D_r & = \dot{\mathbf{R}}^{\mathbf{T}}\mathbf{MD} \\
(\dot{V}_h, D_h) = \sum_{j,r}(\int \dot{V}_j(\ \phi_j \cdot \phi_r))D_r & = \dot{\mathbf{V}}^{\mathbf{T}}\mathbf{MD} \\
(\dot{\Psi}_h, D_h) = \sum_{j,r}(\int \dot{\Psi}_j(\ \phi_j \cdot \phi_r))D_r & = \dot{\mathbf{\Psi}}^{\mathbf{T}}\mathbf{MD} \\
((\dot{A}_1)_h, D_h) = \sum_{j,r}(\int (\dot{A}_1)_j\phi_j \cdot \phi_r))D_r & = (\dot{\mathbf{A}}_{\mathbf{1}})^{\mathbf{T}}\mathbf{MD} \\
((\dot{A}_2)_h, D_h) = \sum_{j,r}(\int (\dot{A}_2)_j(\ \phi_j \cdot \phi_r))D_r & = (\dot{\mathbf{A}}_{\mathbf{2}})^{\mathbf{T}}\mathbf{MD} \\
((\dot{A}_3)_h, D_h) = \sum_{j,r}(\int (\dot{A}_3)_j(\ \phi_j \cdot \phi_r))D_r & = (\dot{\mathbf{A}}_{\mathbf{3}})^{\mathbf{T}}\mathbf{MD} \\
(\dot{p}_h, D_h) = \sum_{j,r}(\int \dot{p}_j(\ \phi_j \cdot \phi_r))D_r & = \dot{\mathbf{p}}^{\mathbf{T}}\mathbf{MD} \\
(\dot{g}_h, D_h) = \sum_{j,r}(\int \dot{g}_j(\ \phi_j \cdot \phi_r))D_r & = \dot{\mathbf{g}}^{\mathbf{T}}\mathbf{MD} \\
(R_h, D_h) = \sum_{i,r}(\int R_i(\phi_i \cdot \phi_r))D_r & = \mathbf{R}^{\mathbf{T}}\mathbf{MD} \\
(V_h, D_h) = \sum_{j,r}(\int V_j(\phi_j \cdot \phi_r))D_r & = \mathbf{V}^{\mathbf{T}}\mathbf{MD} \\
(\Psi_h, D_h) = \sum_{k,r}(\int \Psi_k(\phi_k \cdot \phi_r))D_r & = \mathbf{\Psi}^{\mathbf{T}}\mathbf{MD} \\
((A_1)_h, D_h) = \sum_{m,r}(\int (A_1)_m(\phi_m \cdot \phi_r))D_r & = \mathbf{A}_{\mathbf{1}}^{\mathbf{T}}\mathbf{MD} \\
((A_2)_h, D_h) = \sum_{n,r}(\int (A_2)_n(\phi_n \cdot \phi_r))D_r & = \mathbf{A}_{\mathbf{2}}^{\mathbf{T}}\mathbf{MD} \\
((A_3)_h, D_h) = \sum_{o,r}(\int (A_3)_o(\phi_o \cdot \phi_r))D_r & = \mathbf{A}_{\mathbf{3}}^{\mathbf{T}}\mathbf{MD} \\
(p_h, D_h) = \sum_{s,r}(\int p_s(\phi_s \cdot \phi_r))D_r & = \mathbf{p}^{\mathbf{T}}\mathbf{MD} \\
(g_h, D_h) = \sum_{q,r}(\int g_q(\phi_q \cdot \phi_r))D_r & = \mathbf{g}^{\mathbf{T}}\mathbf{MD} \\
(R_hV_h, D_h) = \sum_{i,j,r}(\int R_iV_j(\phi_i \cdot \phi_j \cdot \phi_r))D_r & \approx (\mathbf{R} \odot \mathbf{V})^{\mathbf{T}}\mathbf{MD} \\
((R^2)_h, D_h) = \sum_{i,r}(\int R_i^2(\phi_i \cdot \phi_i \cdot \phi_r))D_r & \approx (\mathbf{R} \odot \mathbf{R})^{\mathbf{T}}\mathbf{MD} \\
((V^2)_h, D_h) = \sum_{j,r}(\int V_j^2(\phi_j \cdot \phi_j \cdot \phi_r))D_r & \approx (\mathbf{V} \odot \mathbf{V})^{\mathbf{T}}\mathbf{MD} \\
(\nabla\Psi_h, \nabla D_h) = \sum_{k,r}(\int \Psi_k(\nabla\phi_k \cdot \nabla\phi_r))D_r & = \mathbf{\Psi}^{\mathbf{T}}\mathbf{AD} \\
(1, D_h) = \sum_{t,r}(\int 1(\phi_t \cdot \phi_t))D_r & = \mathbf{1}^{\mathbf{T}}\mathbf{MD}
\end{cases} \tag{11}$$

Where $\mathbf{A}$ is the stiffness matrix and $\mathbf{M}$ is the mass matrix and

$$\begin{aligned}
(\dot{R}_h, D_h) &= \frac{1}{\tau}\bigg(-\kappa_V(R_h, D_h) + 2(R_hV_h, D_h) + \frac{\gamma}{\pi\tau}(1_h, D_h)\bigg) \\
&\approx \frac{1}{\tau}\bigg(-\kappa_V(\mathbf{R}^{\mathbf{T}}\mathbf{MD}) + 2(\mathbf{R} \odot \mathbf{V})^{\mathbf{T}}\mathbf{MD} + \frac{\gamma}{\pi\tau}(\mathbf{1}^{\mathbf{T}}\mathbf{MD})\bigg) \\
&= \frac{1}{\tau}\bigg(-\kappa_V(\mathbf{R}^{\mathbf{T}}\mathbf{M}) + 2((\mathbf{R} \odot \mathbf{V}^{\mathbf{T}})\mathbf{M}) + \frac{\gamma}{\pi\tau}(\mathbf{1}^{\mathbf{T}}\mathbf{M})\bigg)\mathbf{D} \\
&= \mathbf{B}_{\mathbf{1}}^{\mathbf{T}}\mathbf{D}
\end{aligned}$$

where $\mathbf{B}_{\mathbf{1}}^{\mathbf{T}} = \frac{1}{\tau}\bigg(-\kappa_V(\mathbf{R}^{\mathbf{T}}\mathbf{M}) + 2(\mathbf{R} \odot \mathbf{V})^T M + \frac{\gamma}{\pi\tau}(\mathbf{1}^{\mathbf{T}}\mathbf{M})\bigg)$. Similarly,

$$\begin{aligned}
(\dot{V}_h, D_h) &= \frac{1}{\tau}\bigg(\kappa_S(g_h, D_h) - (\pi^2\tau^2(R_h^2, D_h)) + (V_h^2, D_h) + \eta_0(1_h, D_h)\bigg) \\
&\approx \frac{1}{\tau}\bigg(k_S(\mathbf{g}^{\mathbf{T}}\mathbf{MD}) - \pi^2\tau^2(\mathbf{R} \odot \mathbf{R})^{\mathbf{T}}\mathbf{MD} + (\mathbf{V} \odot \mathbf{V})^{\mathbf{T}}\mathbf{MD} + \eta_0(\mathbf{1}^{\mathbf{T}}\mathbf{MD})\bigg) \\
&= \frac{1}{\tau}\bigg(k_S(\mathbf{g}^{\mathbf{T}}\mathbf{M}) - \pi^2\tau^2(\mathbf{R} \odot \mathbf{R})^{\mathbf{T}}\mathbf{M} + (\mathbf{V} \odot \mathbf{V})^{\mathbf{T}}\mathbf{M} + \eta_0(\mathbf{1}^{\mathbf{T}}\mathbf{M})\bigg)\mathbf{D} \\
&= \mathbf{B}_{\mathbf{2}}^{\mathbf{T}}\mathbf{D}
\end{aligned}$$

So,

$$\begin{aligned}(\ddot{R}_h, D_h) &= \frac{1}{\tau}\Big(-\kappa_V(\dot{R}_h, D_h) + 2(V_h\dot{R}_h, D_h) + 2(R_h\dot{V}_h, D_h)\Big) \\ &\approx \frac{1}{\tau}\Big(-\kappa_V \mathbf{B_1^T M} + 2(\mathbf{V}\odot\mathbf{B_1})^{\mathbf{T}}\mathbf{M} + 2(\mathbf{R}\odot\mathbf{B_2})^{\mathbf{T}}\mathbf{M}\Big)\mathbf{D} \\ &= \mathbf{B_3^T D}\end{aligned}$$

and

$$\begin{aligned}(A_4, D_h) &= -\frac{3}{2}(\nabla(A_2)_h, \nabla D_h) - \frac{3}{2}(\nabla R_h, \nabla D_h) - \frac{1}{\nu^2}(\ddot{R}_h, D_h) - \frac{1}{\nu}(\dot{R}_h, D_h) \\ &\approx \Big(-\frac{3}{2}(\mathbf{A_2^T AD}) - \frac{3}{2}(\mathbf{R^T AD}) - \frac{1}{\nu^2}(\mathbf{B_3^T D}) - \frac{1}{\nu}(\mathbf{B_1^T D})\Big)\end{aligned}$$

Substituting (11) and (10) into (9), we get the system of ODEs

$$\begin{cases}\dot{\mathbf{R}}^{\mathbf{T}}\mathbf{MD} &= \frac{1}{\tau}\Big(-\kappa_V\mathbf{R^T MD} + 2((\mathbf{R}\odot\mathbf{V})^{\mathbf{T}}\mathbf{MD}) + \frac{\gamma}{\tau\pi}(\mathbf{1^T M})\mathbf{D}\Big) \\ \dot{\mathbf{V}}^{\mathbf{T}}\mathbf{MD} &= \frac{1}{\tau}\Big(\kappa_s(\mathbf{g^T M})\mathbf{D} - (\pi^2\tau^2(\mathbf{R}\odot\mathbf{R})^{\mathbf{T}}\mathbf{MD}) + ((\mathbf{V}\odot\mathbf{V})^{\mathbf{T}}\mathbf{MD}) + \eta_0(\mathbf{1^T M})\mathbf{D}\Big) \\ \dot{\mathbf{\Psi}}^{\mathbf{T}}\mathbf{MD} &= \nu(-\mathbf{\Psi^T MD} + \mathbf{A_1^T MD}) \\ \dot{\mathbf{A}}_1^{\mathbf{T}}\mathbf{MD} &= \nu\Big(-\mathbf{A_1^T MD} + \mathbf{A_2^T MD} + \tfrac{3}{2}(\mathbf{\Psi^T AD})\Big) \\ \dot{\mathbf{A}}_2^{\mathbf{T}}\mathbf{MD} &= \nu(\mathbf{A_3^T MD} - \mathbf{A_2^T MD}) \\ \dot{\mathbf{A}}_3^{\mathbf{T}}\mathbf{MD} &= \nu\Big(\tfrac{3}{2}(\mathbf{A_2^T AD}) - \tfrac{3}{2}(\mathbf{R^T AD}) + \tfrac{1}{\nu^2}(\mathbf{B_3^T D}) - \tfrac{1}{\nu}(\mathbf{B_1^T D}) - \mathbf{A_3^T MD}\Big) \\ \dot{\mathbf{p}}^{\mathbf{T}}\mathbf{MD} &= \alpha(\mathbf{\Psi^T MD} - \mathbf{p^T MD}) \\ \dot{\mathbf{g}}^{\mathbf{T}}\mathbf{MD} &= \alpha(\mathbf{p^T MD} - \mathbf{g^T MD})\end{cases}$$

which must hold for every $\mathbf{D}$.
This ODE system is of the form

$$\mathbf{N}\dot{\mathbf{y}} = \mathbf{f}(\mathbf{y})$$

where

$$\mathbf{N} = \begin{bmatrix}\mathbf{M} & & \\ & \ddots & \\ & & \mathbf{M}\end{bmatrix}$$

and

$$\mathbf{y} = \begin{bmatrix}\mathbf{R}\\ \mathbf{V}\\ \mathbf{\Psi}\\ \mathbf{A_1}\\ \mathbf{A_2}\\ \mathbf{A_3}\\ \mathbf{p}\\ \mathbf{g}\end{bmatrix}$$

and

$$\mathbf{f}(\mathbf{y}) = \begin{bmatrix} \frac{1}{\tau}\big(-\kappa_V \mathbf{MR} + 2(\mathbf{M}(\mathbf{R}\odot\mathbf{V})) + \frac{\gamma}{\tau\pi}(\mathbf{M1})\big) \\ \frac{1}{\tau}\big(\kappa_s(\mathbf{Mg}) - (\pi^2\tau^2\mathbf{M}(\mathbf{R}\odot\mathbf{R})) + (\mathbf{M}(\mathbf{V}\odot\mathbf{V})) + \eta_0(\mathbf{M1})\big) \\ \nu(-\mathbf{M\Psi} + \mathbf{MA_1}) \\ \nu\Big(-\mathbf{MA_1} + \mathbf{MA_2} + \frac{3}{2}(\mathbf{A\Psi})\Big) \\ \nu(\mathbf{MA_3} - \mathbf{MA_2}) \\ \nu\Big(\frac{3}{2}\mathbf{AA_2} - \frac{1}{v}\mathbf{B_1} + \frac{1}{v^2}\mathbf{B_3} - \frac{3}{2}\mathbf{AR} - \mathbf{MA_3}\Big) \\ \alpha(\mathbf{M\Psi} - \mathbf{Mp}) \\ \alpha(\mathbf{Mp} - \mathbf{Mg}) \end{bmatrix}$$

If we use a Eulerian discretization of time as in the case of the time-dependent heat equation, we can find solutions of the form

$$\mathbf{y_{k+1}} = \mathbf{y_k} + \Delta t \mathbf{N^{-1}} \mathbf{f}(\mathbf{y_k})$$

We solved this system with MATLAB's ODE45 function. The variable that we plot in our numerical experiments is Ψ.

6.4 Synchrony

We can determine the *within-population* dynamics by a conformal map derived by Montbrio.

$$\begin{cases} Z = \frac{1-W^*}{1+W^*} \\ W = \pi\tau R + iV \end{cases}$$

We can follow a point in mesh through a simulation and plot the R,V and corresponding $|Z|$-values which fall between 0 and 1, 0 being a low synchrony level and 1 being a high level [4].

7 Bifurcation Analysis

The mathematician Alan Turing developed the concept of Turing instability analysis in 1952 to study emergence of patterns in spatio-temporal systems.

7.1 Planar Surface

We can do a stability analysis of the system in the manner of Coombes [4] by first finding the homogeneous steady state and perturbing from this value. More specifically, the steady state will be given by solving the equation:

$$\begin{aligned} -\kappa_V R_0 + 2R_0V_0 + \frac{\gamma}{\pi\tau} &= 0 \\ \eta_0 + V_0^2 - \pi^2\tau^2 R_0^2 &= 0 \end{aligned}$$

for steady state

$$(0, 0, R_0, V_0)$$

and then applying perturbations of the form

$$(U(\boldsymbol{r}, t), \Psi(\boldsymbol{r}, t), V(\boldsymbol{r}, t)) = (0, 0, R_0, V_0) + \epsilon(\overline{U}, \overline{\Psi}, \overline{R}, \overline{V}) e^{\lambda t} \boldsymbol{e}^{i\boldsymbol{\xi}\cdot\boldsymbol{r}} \tag{12}$$

If this equation is substituted into the brain wave equation, then, we get

$$\left[\left(1 + \frac{\lambda}{\nu}\right)^2 + \frac{3}{2}\xi^2\right]^2 \overline{\Psi} = -\left[\frac{\lambda}{\nu}\left(1 + \frac{\lambda}{\nu}\right) + \frac{3}{2}\xi^2\right]\overline{R}$$

where $\xi = |\boldsymbol{\xi}|$.
Linearizing the dynamics of (R, V), we get

$$A(\lambda)\begin{bmatrix}\overline{R} \\ \overline{V}\end{bmatrix} = \begin{bmatrix}0 \\ \kappa_s \overline{U}\end{bmatrix}$$
$$A(\lambda) = \tau\lambda I_2 - J$$

$$J = \begin{bmatrix}-\kappa_V + 2V_0 & 2R_0 \\ -2\pi^2\tau^2 R_0 & 2V_0\end{bmatrix}$$

Now, use Cramer's Rule to obtain expression for

$$\overline{R} = \frac{1}{|A(\lambda)|}\begin{vmatrix}0 & -2R_0 \\ \kappa_s\overline{U} & \tau\lambda - 2V_0\end{vmatrix} = \frac{2\kappa_s R_0}{|A(\lambda)|\left(1 + \frac{\lambda}{\alpha}\right)^2}\overline{\Psi}$$

We used the fact that

$$\left(1 + \frac{\lambda}{\alpha}\right)^2 \overline{U} = \overline{\Psi} \tag{13}$$

Substitute back into original equation and ask for a nontrivial solution for $\overline{\Psi}$. This means that

$$\mathcal{E}(\lambda, \xi) = 0$$

where

$$\mathcal{E}(\lambda, \xi) = |A(\lambda)|\left(1 + \frac{\lambda}{\alpha}\right)^2\left[\left(1 + \frac{\lambda}{\nu}\right)^2 + \frac{3}{2}\xi^2\right]^2 + 2\kappa_s R_0\left[\frac{\lambda}{\nu}\left(1 + \frac{\lambda}{\nu}\right) + \frac{3}{2}\xi^2\right]$$

To do bifurcation analysis on a manifold, we would perform a similar analysis, replacing the quantity $e^{i\boldsymbol{\xi}\cdot\mathbf{r}}$ by an eigenfunction of the surface Laplacian in our ansatz (12). Many of the same tools used for analysis of pattern formation of reaction diffusion equations on surfaces, for which there is a large body of literature already extant, can be applied to studying pattern formation of neural fields as well. That is our aim.

7.2 Sphere

We are interested in investigating the effect of surface geometry on pattern formation on the human cortex. To study the dynamics, we applied techniques used in the study of nonlinear reaction diffusion systems on arbitrary surfaces [7]. These techniques involve conducting numerical analysis of the Laplace-Beltrami operator on surface meshes. Our final goal is to understand the possible patterns in the system and which solutions are stable and unstable.
Because the Laplace-Beltrami operator acting on an arbitrary surface is Hermitian, we can compute the spectral basis for an emerging pattern.
We can substitute a potential pattern into a linearized system to obtain an explicit form of the pattern. Thereby, we can define a bifurcation point in terms of spectral eigenvalues and system parameters.
We first perform bifurcation analysis near homogeneity through spectral analysis of the Laplace-Beltrami operator. From this operator, we can obtain eigenmodes and eigenvalues for the operator.
To compute the *Laplacian basis eigenfunctions*, we solve the eigenvalue problem

$$\nabla^2 \phi_k = -\lambda_k \phi_k,$$

In numerical terms, we discretize the eigenvalue problem as

$$\boldsymbol{A}\boldsymbol{b}_k = \lambda_k \boldsymbol{M}\boldsymbol{b}_k$$

where $\boldsymbol{b}_k$ is a discrete vector representation of ϕ_k, $\boldsymbol{A}$ is the stiffness matrix and $\boldsymbol{M}$ is the mass matrix.
Martin performed a numerical bifurcation analysis on the torus using a *pseudo-arclength continuation scheme*. One observation of note is that, on the torus, as the major curvature radius decreases to a certain point, a stable bump solution evolves into a stable ring solution.

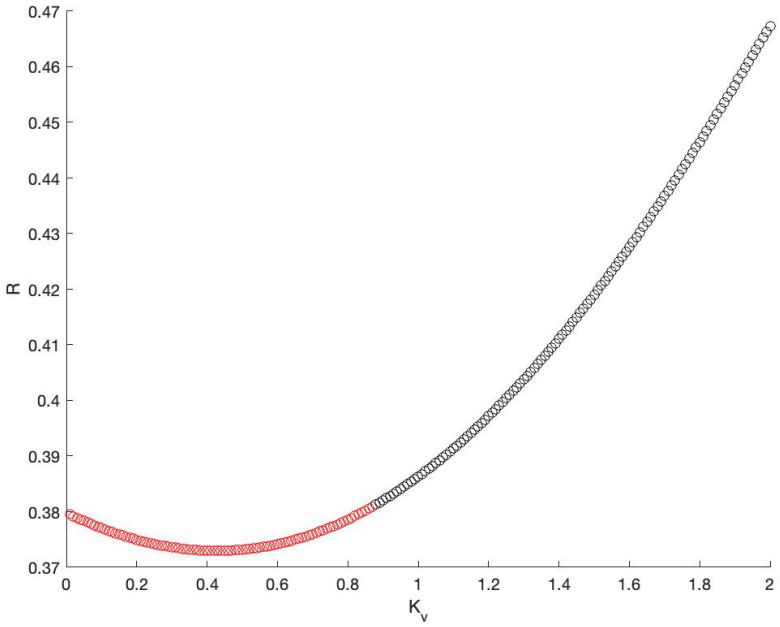
8 Plots

In following simulations, we use parameters

$$\begin{aligned} \nu &= 1 \\ \eta_0 &= 3 \\ \Delta &= 0.5 \\ \alpha &= 1 \\ \kappa_\nu &= 0.5 \textbf{ or } 3 \\ \kappa_s &= 1 \\ \tau &= 1 \end{aligned}$$

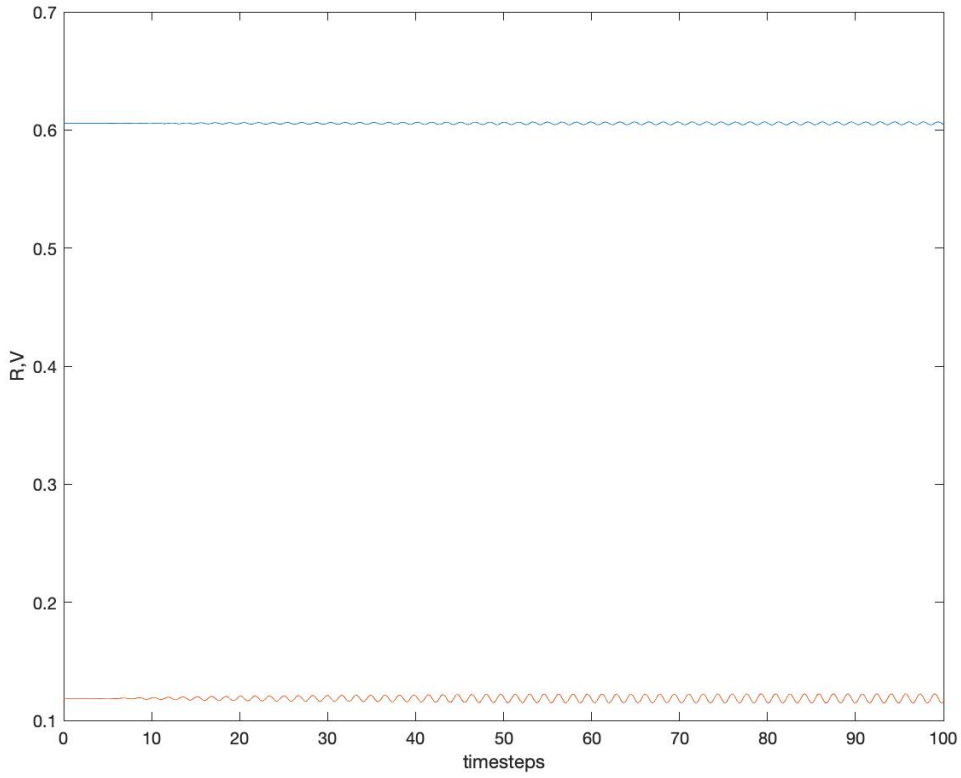
for the neural mass model and the neural field models. The neural mass model is system of 4 ODEs.

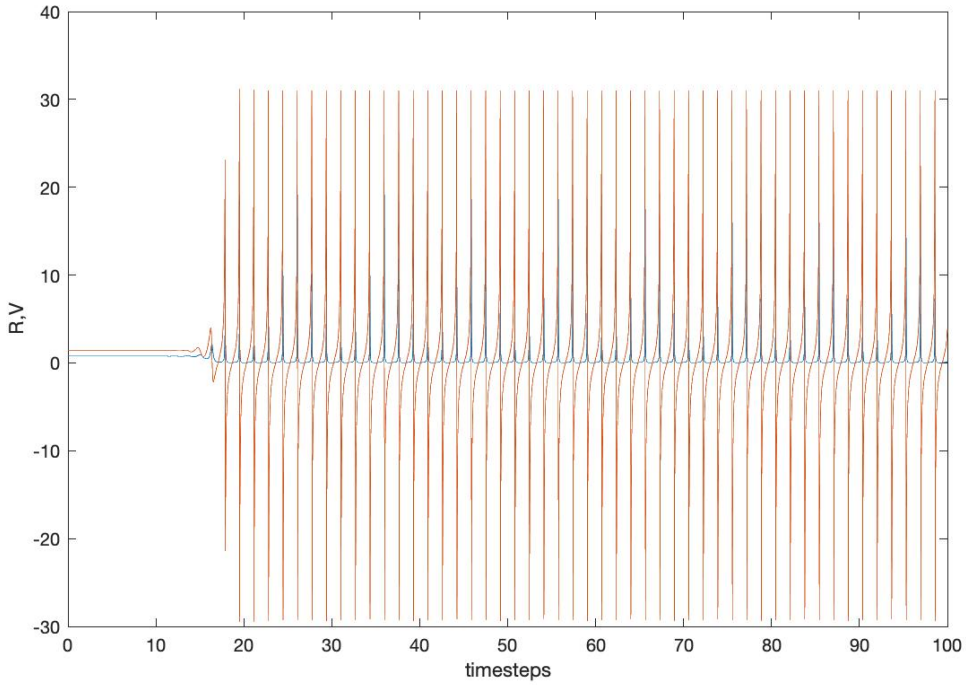
8.1 Bifurcation Analysis



Where the graph above changes from red (stable) to black (unstable) is a Hopf bifurcation point for the neural mass model.

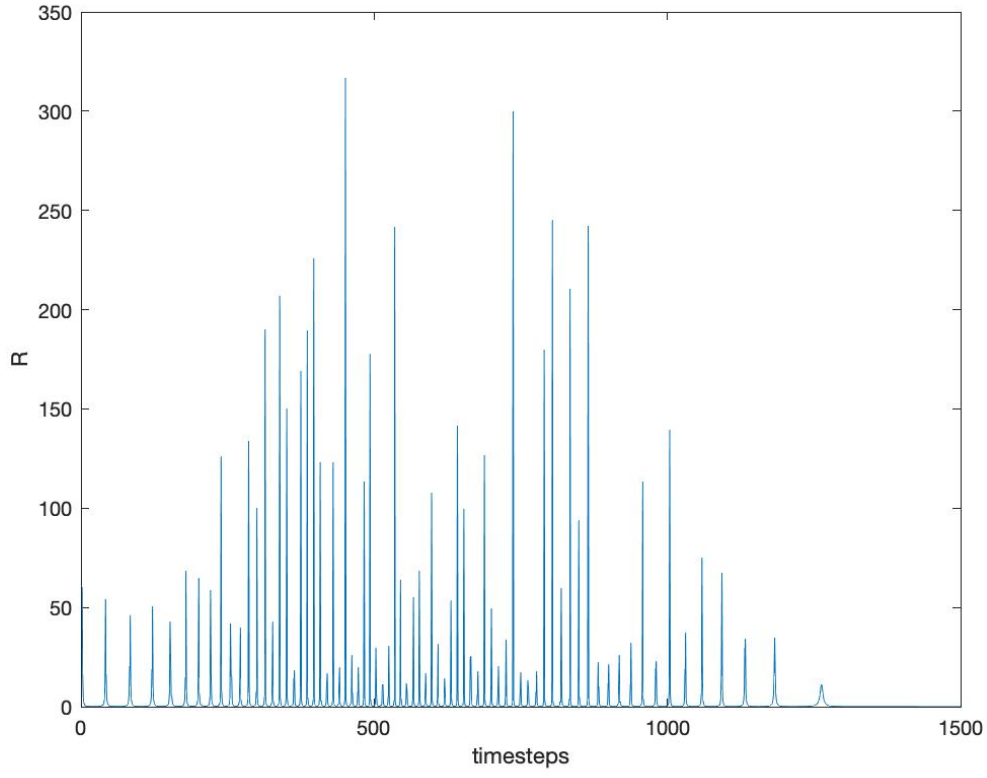
8.1.1 Neural Mass Model

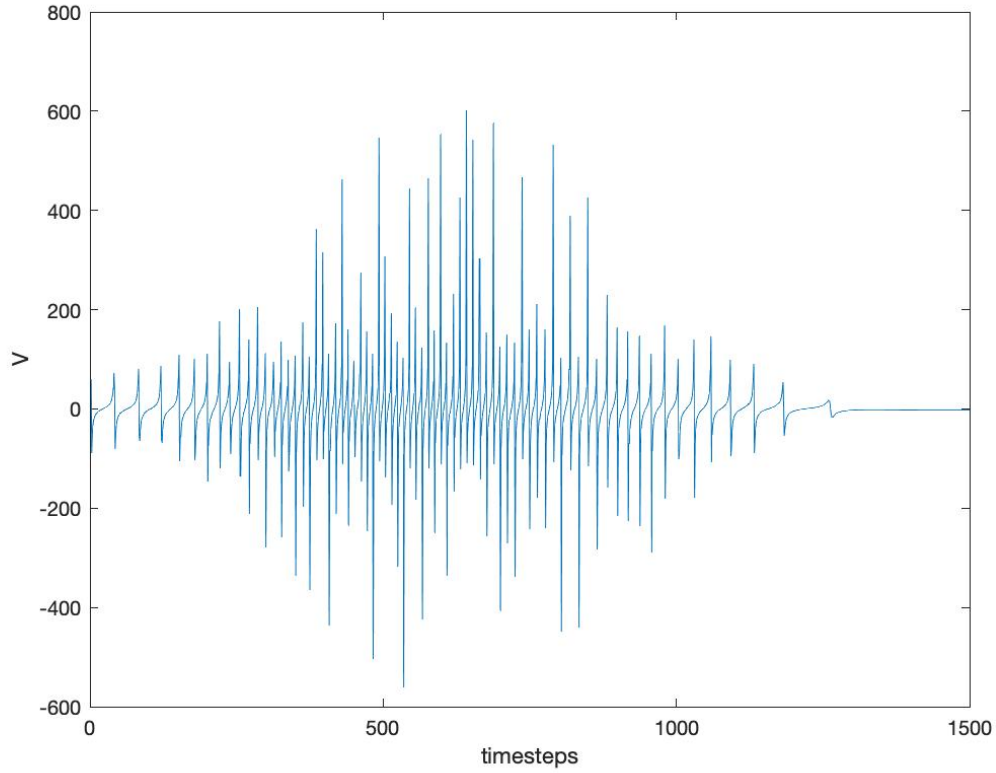




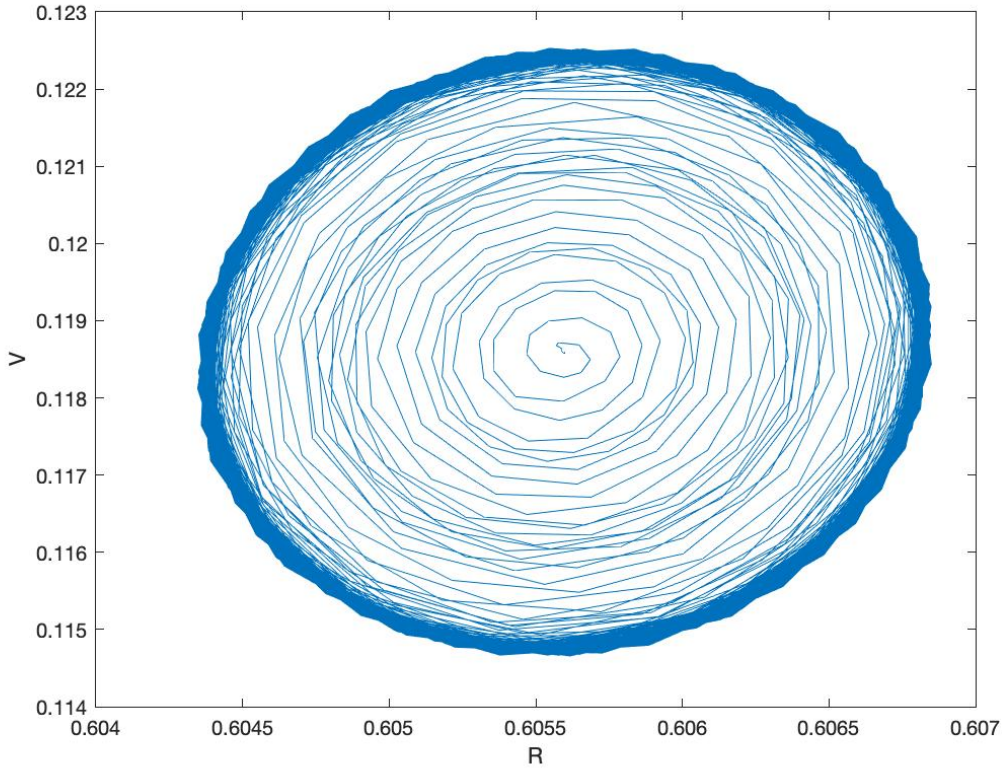
The first plot above shows R (blue) and V (orange) converging to steady state (when $\kappa_V = 0.5$). The second plot show R and V in unstable state (when $\kappa_V = 3$).

8.1.2 2D Neural Field Model

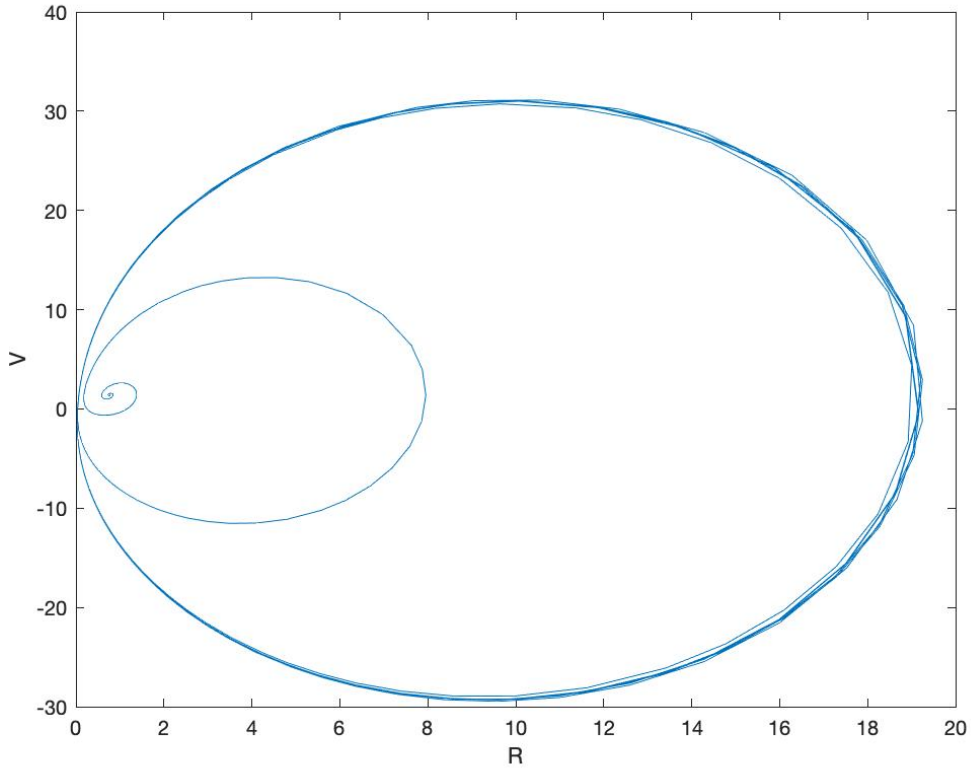




Graphs above are for R and V in the 2D neural field (for a single point near the pole of the sphere) in stable region where $\kappa_V = 0.5$. We can see both approach a steady state occurring between 1000 and 1500 timesteps.



The plot above shows a limit cycle resulting from stable flow ($k_V = 0.5$).

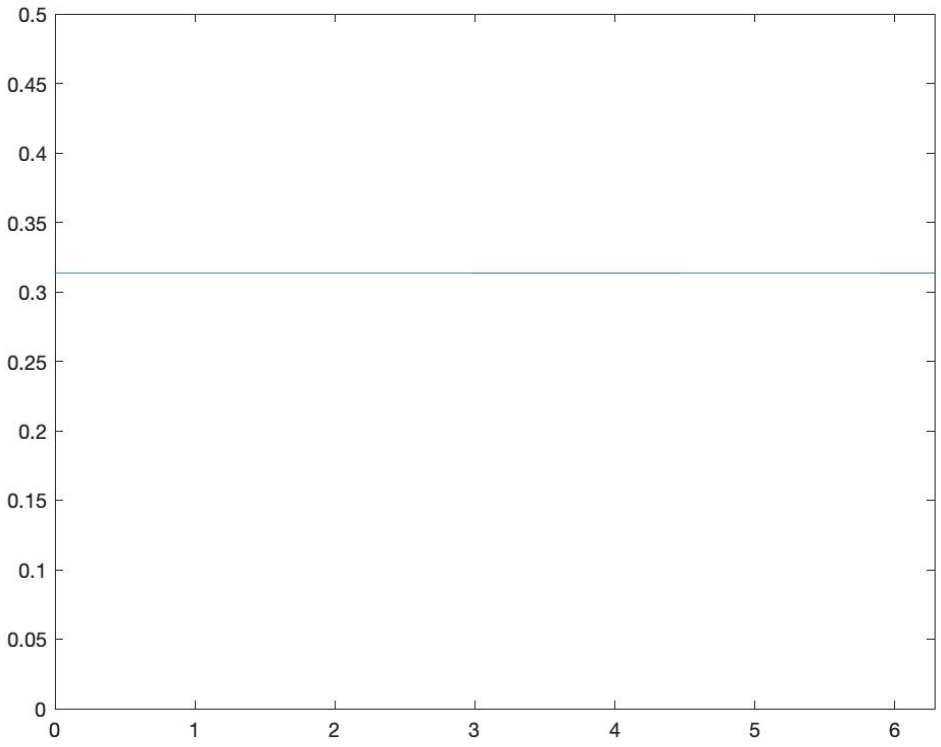


The plot above shows a limit cycle resulting from unstable flow ($k_V = 3$). We can see the width has greatly expanded.

8.2 Synchronization

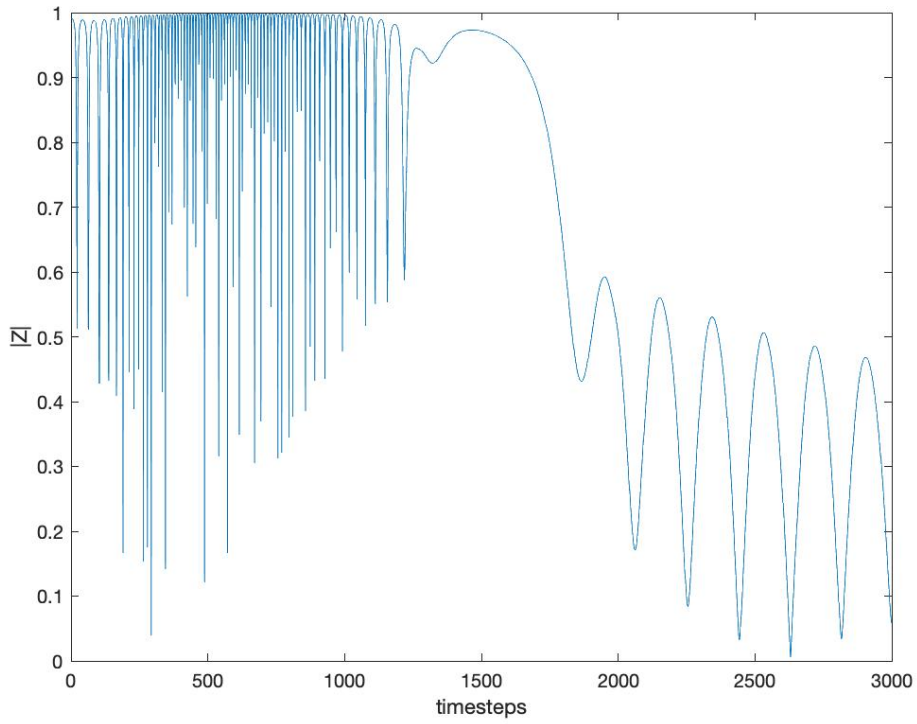
Coombes found that by increasing the gap junction coupling strength of a neural field on a planar surface, the level of synchronisation across the tissue, as measured by Z, increases [5]. The peak synchrony value is almost 1 and the minimum value is roughly 0.80, which seems to support the hypothesis that gap junction coupling lends to more synchronous activity.

8.2.1 Neural Mass Model



In stable region of R and V, there is a nearly constant synchrony in the neural mass model.

8.2.2 2D Neural Field Model



graph shows $|Z|$ for stable regime of R and V. We observe significant synchrony existing before the break down at around 1500 time steps with spacing of 0.01.

9 Numerical Experiments

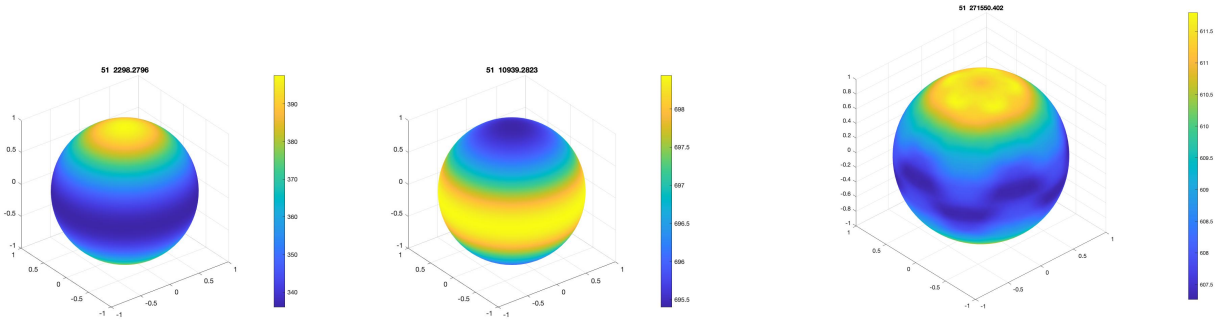
We ran simulations on three surfaces: a planar region with periodic boundary conditions (in progress), a sphere, and a neuroimaging sample of the human brain using MATLAB and FreeFEM++.

The neuroimaging data is a down-sampled mesh of one hemisphere of the brain obtained from the Research Imaging Institute, University of Texas Health Science Center at San Antonio and the mesh was created with FreeSurfer 5.2.0. [3]

9.1 Planar Surface

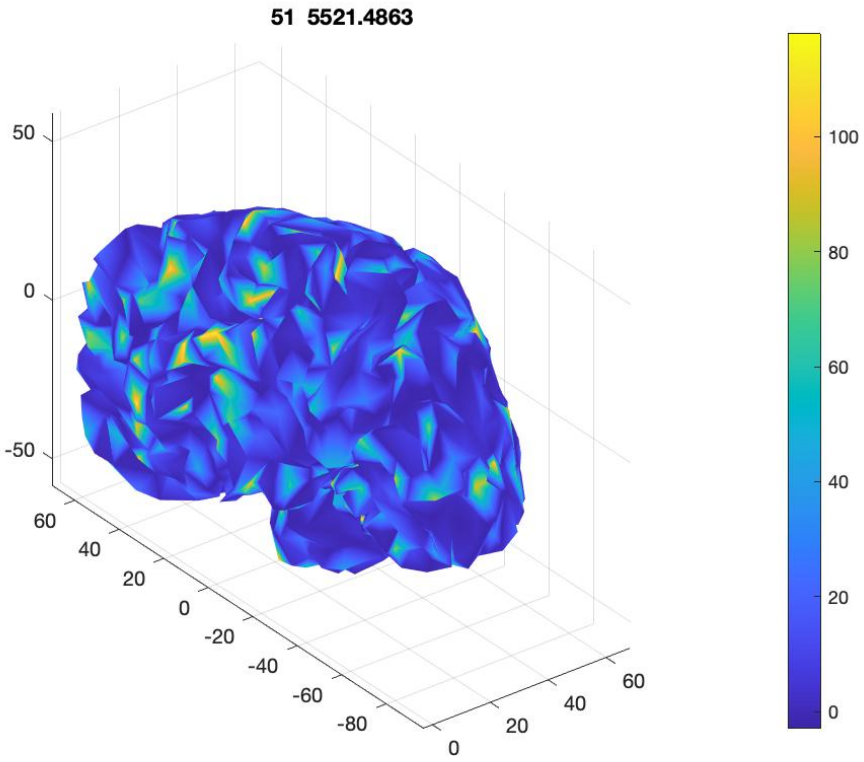
In progress.

9.2 Sphere



The first two plots (left to right) show the typical bump and ring patterns we see in the simulations. The last one shows the emergence of five-leaf clover pattern within a bump, indicating within-population synchrony.

9.3 MRI



We are still running simulations.

10 Conclusions

We observe the curvature of sphere seems to allow for more local synchrony as compared with synchrony in the neural mass model, which stays nearly constant.

Simulations show key patterns seems to be a ring around the equator and bumps at the poles.

11 Future Research

It could be beneficial to study in greater detail the dynamics of the bump patterns (localized states) we obtain from the PDE model posed on the sphere, for which there is a growing literature. For example, Rozada, Ruuth, and Ward investigate the existence and stability of multi-spot formation on the unit sphere for the Brusselator reaction-diffusion [19,20]. Additionally, we would like to gauge the discretization error in the model by comparing with an analytical solution on the sphere, if possible.

12 Sources

[1] Bick, C., Goodfellow, M., Laing, C. R., amp; Martens, E. A. (2020). Understanding the dynamics of biological and neural oscillator networks through exact mean-field reductions: a review. The Journal of Mathematical Neuroscience, 10(1). https://doi.org/10.1186/s13408-020-00086-9

[2] Bressloff, P. C. (2014). Waves in neural media: from single neurons to neural fields. Springer.

[3] Brain for Blender. Brainder. (2018, March 29). https://brainder.org/research/brain-for-blender/.

[4] Byrne, Á., Ross, J., Nicks, R., amp; Coombes, S. (2020). Mean-field models for EEG/MEG: from oscillations to waves. https://doi.org/10.1101/2020.08.12.246256

[5] Coombes, S., amp; Byrne, Á. (2018). Next Generation Neural Mass Models. Nonlinear Dynamics in Computational Neuroscience, 1–16. https://doi.org/10.1007/978-3-319-71048-8$_1$

[6] Dayan, P., amp; Abbott, L. (2014). Theoretical Neuroscience: Computational and Mathematical Modeling of Neural Systems. MIT Press.

[7] Dhillon, D. S., Milinkovitch, M. C., amp; Zwicker, M. (2017). Bifurcation Analysis of Reaction Diffusion Systems on Arbitrary Surfaces. Bulletin of Mathematical Biology, 79(4), 788–827. https://doi.org/10.1007/s11538-017-0255-8

[8] Dziuk, G., amp; Elliott, C. M. (2013). Finite element methods for surface PDEs. Acta Numerica, 22, 289–396. https://doi.org/10.1017/s0962492913000056
[9] Gerstner, W. (n.d.). Neuronal Dynamics. Neuronal Dynamics - a neuroscience textbook by Wulfram Gerstner, Werner M. Kistler, Richard Naud and Liam Paninski. https://neuronaldynamics.epfl.ch/.

[10] Golomb, D. (n.d.). Neuronal synchrony measures. Scholarpedia.

Izhikevich, E. M. (2014). Dynamical Systems in Neuroscience: the Geometry of Excitability and Bursting. MIT Press.

[12] Jirsa, V. K., Jantzen, K. J., Fuchs, A., amp; Kelso, J. A. S. (2001). Neural Field Dynamics on the Folded Three-Dimensional Cortical Sheet and Its Forward EEG and MEG. Lecture Notes in Computer Science, 286–299. https://doi.org/10.1007/3-540-45729-13$0$

[13] Johnson, C. (2009). Numerical solution of partial differential equations by the finite element method. Dover Publications.

[14] Laing, C. R., amp; Troy, W. C. (2003). PDE Methods for Nonlocal Models. SIAM Journal on Applied Dynamical Systems, 2(3), 487–516. https://doi.org/10.1137/030600040

[15] Martin, R. (n.d.). Collocation techniques for solving neural field models on complex cortical geometries (dissertation).

[16] Montbrió, E., Pazó, D., amp; Roxin, A. (2015). Macroscopic Description for Networks of Spiking Neurons. Physical Review X, 5(2). https://doi.org/10.1103/physrevx.5.021028

[17] Ott, E., amp; Antonsen, T. M. (2008). Low dimensional behavior of large systems of globally coupled oscillators. Chaos: An Interdisciplinary Journal of Nonlinear Science, 18(3), 037113. https://doi.org/10.1063/1.2930766

[18] Rognes, M. E., Ham, D. A., Cotter, C. J., amp; McRae, A. T. (2013). Automating the solution of PDEs on the sphere and other manifolds in FEniCS 1.2. Geoscientific Model Development, 6(6), 2099–2119. https://doi.org/10.5194/gmd-6-2099-2013

[19] Rozada, I., Ruuth, S. J., amp; Ward, M. J. (2014). The Stability of Localized Spot Patterns for the Brusselator on the Sphere. SIAM Journal on Applied Dynamical Systems, 13(1), 564–627. https://doi.org/10.1137/130934696

[20] Trinh, P. H., amp; Ward, M. J. (2016). The dynamics of localized spot patterns for reaction-diffusion systems on the sphere. Nonlinearity, 29(3), 766–806. https://doi.org/10.1088/0951-7715/29/3/766

[21] Visser, S., Nicks, R., Faugeras, O., amp; Coombes, S. (2017). Standing and travelling waves in a spherical brain model: The Nunez model revisited. Physica D: Nonlinear Phenomena, 349, 27–45. https://doi.org/10.1016/j.physd.2017.02.017