Anisotropic stress-assisted reaction-diffusion and poroelasticity

Candidate Number 1008097
Mathematics (3 or 4 years)

Hilary Term
2018

Abstract

In this project, we set out to explore two existing models: that of poroelastic materials under stress, (e.g. a sponge filled with water being pressed) and that of anisotropic reaction-diffusion (e.g. propagation of electrical signals through heart tissue). We will use the finite-element method (implemented in the python library, FEniCS) to solve the resultant PDEs for different boundary conditions, and finally, we will combine the two models together to describe a poroelastic material whose internal stresses are modified by diffusion within it.

1 Reaction-Diffusion

1.1 Introducing the model

Reaction-diffusion is the name given to a system where there are one or more species *diffusing* through a medium and *reacting* with that medium—or each other—in some way. We will focus on two main cases; two reacting chemicals, and the diffusion of electrochemical signals through cardiac tissue.

1.2 Derivation of equations

The general case for species diffusing and reacting is quite simple to describe. Consider a fixed domain, $\Omega \subset \mathbb{R}^d$ over which we define a concentration $c(t) : \Omega \to \mathbb{R}$. We assume that this domain has a **diffusion tensor** $\mathbf{D} : \Omega \to \mathbb{R}^{d\times d}$, which contains information on how species diffuse through it in different directions. For isotropic domains (i.e. domains that have the same diffusion characteristics in every direction), $\mathbf{D} = D\mathbf{I}$, D a diffusion constant.

Ω has boundary $\partial\Omega = \Gamma_D \cup \Gamma_N$, as before split into two disjoint sections on which we prescribe Dirichlet and Neumann boundary conditions respectively:

$$c = c_D \qquad \text{on } \Gamma_D, \tag{1}$$

$$\mathbf{D}\nabla c \cdot \boldsymbol{n} = 0 \qquad \text{on } \Gamma_N. \tag{2}$$

In the absence of any reaction, the equation describing the system is simply

$$\frac{\partial}{\partial t} c = \operatorname{div}\left(\mathbf{D}\nabla c\right) \quad \text{in } \Omega.^1$$

In the most general sense (as 'reaction' includes a wide range of phenomena), the reaction-diffusion equation is

$$\frac{\partial}{\partial t} c - \operatorname{div}\left(\mathbf{D}\nabla c\right) = r(c) \quad \text{in } \Omega,{}^1 \tag{3}$$

where r is a **reaction function** that can be constructed to fit any system. If there is more than one species diffusing and reacting, change (3) to

$$\frac{\partial}{\partial t} \boldsymbol{c} - \mathbf{div}\left(\mathbf{D}\nabla \boldsymbol{c}\right) = \boldsymbol{r}(\boldsymbol{c}) \quad \text{in } \Omega,{}^1 \tag{4}$$

where

$$\boldsymbol{c} = \begin{pmatrix} c_1 \\ c_2 \\ \vdots \\ c_n \end{pmatrix}$$

is a vector of concentrations, and c_i is the concentration of the i-th species at a point in space and time.

1.2.1 Weak form

As before, in order to use the finite element method to solve this equation, we must write it in weak form. Starting with the scalar case (3), we multiply by a test function d, and integrate over Ω:

$$\int_\Omega \frac{\partial}{\partial t} c d \,\mathrm{d}V + \int_\Omega \operatorname{div}(\mathbf{D}\nabla c) d \,\mathrm{d}V = \int_\Omega r(c) d \,\mathrm{d}V \quad \forall d.$$

We then integrate the middle integral by parts, moving derivatives onto the test function:

$$\int_\Omega \frac{\partial}{\partial t} c d \,\mathrm{d}V + \int_\Omega \mathbf{D}\nabla c \cdot \nabla d \,\mathrm{d}V - \int_{\partial\Omega} \mathbf{D}\nabla c \cdot \boldsymbol{n}\, d \,\mathrm{d}s = \int_\Omega r(c) d \,\mathrm{d}V \quad \forall d.$$

Notice that the term inside the integral over $\partial\Omega$ is zero on Γ_N according to (2), so we can rewrite the above as

$$\int_\Omega \frac{\partial}{\partial t} c d \,\mathrm{d}V + \int_\Omega \mathbf{D}\nabla c \cdot \nabla d \,\mathrm{d}V - \int_{\Gamma_D} \mathbf{D}\nabla c \cdot \boldsymbol{n}\, d \,\mathrm{d}s = \int_\Omega r(c) d \,\mathrm{d}V \quad \forall d,$$

and as a final step, we pick d such that it is 0 everywhere on Γ_D, removing that term entirely and giving us the final **general reaction-diffusion weak form**

$$\int_\Omega \frac{\partial}{\partial t} c d \,\mathrm{d}V + \int_\Omega \mathbf{D}\nabla c \cdot \nabla d \,\mathrm{d}V = \int_\Omega r(c) d \,\mathrm{d}V \quad \forall d \text{ s.t. } d = 0 \text{ on } \Gamma_D. \tag{5}$$

To find the vector form, simply replace c and d with $\boldsymbol{c}$ and $\boldsymbol{d}$ respectively.

1.2.2 Time discretisation

We are not finished yet, however. The concentration c varies over time, so we must employ **time discretisation**.

We split the time domain $[0, T)$, with T a maximum time, into $N + 1$ time steps, separated by a **timestep** $\Delta t = T/(N + 1)$. Let:

$$t^n = n\Delta t, \quad n = 0, 1, \ldots, N + 1$$

[1] Ricardo Ruiz-Baier, private conversation.

be the discrete time values we consider, and let c^n be our numerical approximation of $c(t^n)$. We will employ the following approximation of the time derivative,

$$\frac{\partial}{\partial t}c(t^n) \approx \frac{c^{n+1} - c^n}{\Delta t},$$

giving us our final, time-discretised weak form problem:

$$\frac{1}{\Delta t}\int_\Omega c^{n+1} d \, \mathrm{d}V + \int_\Omega \mathbf{D}\nabla c^{n+1} \cdot \nabla d \, \mathrm{d}V = \int_\Omega r(c^n) d \, \mathrm{d}V + \frac{1}{\Delta t}\int_\Omega c^n d \, \mathrm{d}V, \qquad \forall d \text{ s.t. } d = 0 \text{ on } \Gamma_D. \tag{6}$$

Note that we have a **semi-implicit** scheme here: this scheme is implicit in the diffusion terms, but explicit in the reaction terms, as you can see by c^{n+1} appearing with $\mathbf{D}$ the diffusion tensor, but c^n appearing inside the reaction function r.

We are now ready to use the finite element method to numerically solve this system. The finite element method is not taught at an undergraduate level, but full knowledge of its inner workings is not required to understand this project, only the knowledge that it is a method to numerically solve a system of equations like the ones we are using here. For a whistle-stop tour that will give a non-rigorous introduction to the finite element method, see appendix A.

1.3 Schnackenberg computation

For our first test problem, we will consider a system of two diffusing species in a domain $\Omega = (0,1)^2$ that react with each other. We will call the concentrations of the two species $u(t)$ and $v(t)$. The system is given to us as:

$$\left.\begin{array}{rcl} \frac{\partial}{\partial t}u &=& c\nabla^2 u + d(a - u + u^2 v), \\ \frac{\partial}{\partial t}v &=& \nabla^2 v + d(b - u^2 v), \end{array}\right\} \text{ in } \Omega, \tag{7}$$

where a, b, c, and d are given constants. We shall write this in the form of (4), letting $\boldsymbol{u} = (u, v)^T$:

$$\frac{\partial}{\partial t}\boldsymbol{u} - \begin{pmatrix} c & 0 \\ 0 & 1 \end{pmatrix} \nabla^2 \boldsymbol{u} = d \begin{pmatrix} a - u + u^2 v \\ b - u^2 v \end{pmatrix}, \tag{8}$$

where, as you can see, we have set

$$\mathbf{D} = \begin{pmatrix} c & 0 \\ 0 & 1 \end{pmatrix}, \qquad \boldsymbol{r}(\boldsymbol{u}) = d \begin{pmatrix} a - u + u^2 v \\ b - u^2 v \end{pmatrix}.$$

We impose no-flux boundary conditions—that is,

$$\left.\begin{aligned} \nabla u \cdot \boldsymbol{n} &= 0, \\ b\nabla v \cdot \boldsymbol{n} &= 0, \end{aligned}\right\} \text{ on } \partial\Omega. \tag{9}$$

We are given the constant values

$$\begin{aligned} a &= 0.1305, \\ b &= 0.7695, \\ c &= 0.05, \\ d &= 170, \end{aligned}$$

and we are given the initial conditions

$$\begin{aligned} u(0) &= a + b + 0.001 \exp\left(-100\left((x - 1/3)^2 + (y - 1/2)^2\right)\right), \\ v(0) &= \frac{b}{(a+b)^2}. \end{aligned}$$

We take the timestep to equal:

$$\Delta t = \frac{1}{64^2} \approx 0.00122$$

1.3.1 Results of computation

For all code used in this project, please refer to appendix B.

We present the numerical solution of this equation. As you can see in figure 1, the concentration $u(t)$ of the first species starts out almost completely homogeneous. After 0.244 seconds, these ripple-like features start to emerge. This is very interesting, as there was nothing in the initial condition to suggest such features, they have cropped up seemingly out of nowhere. In the next two images we see the ripples themselves begin to break up, into spots of high concentration surrounded by areas of practically zero concentration at all. In the final picture (which is from a time relatively long after the first 4 figures), we see a steady state begin to emerge: ‘islands’ of concentration surrounded by nothing else. The spots of concentration appear to form circles centred around a certain spot, which corresponds to the location of the initial concentration.

Figure 2 shows that the concentration $v(t)$ almost exactly fits the opposite of $u(t)$, and reaches an equilibrium at the same time.

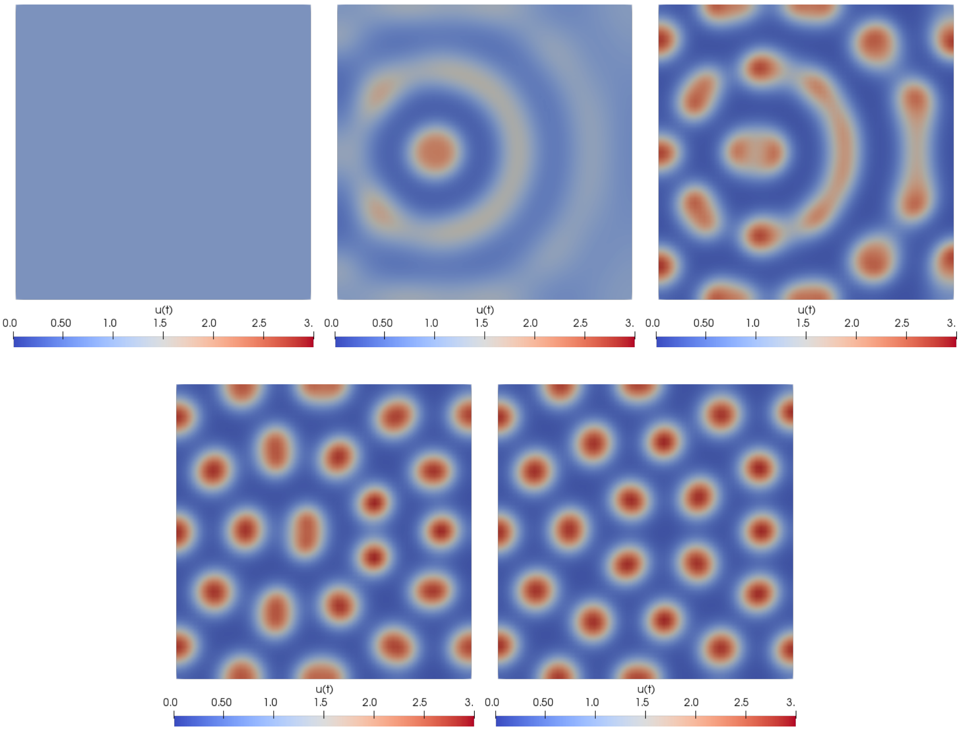

Figure 1: $u(t)$ at $t = 0, 0.244, 0.366, 0.732, 1.953$.

1.4 Karma computation

For our second example, we will use the system discussed in [2], which is a model for propagation of electrical potential in cardiac (heart) tissue. Karma gives the relevant model equations as:

$$\frac{\partial}{\partial t} E = D_0 \operatorname{div}(\nabla E) + \frac{1}{\tau_E} f(E, n), \tag{10}$$

$$\frac{\partial}{\partial t} n = \frac{1}{\tau_n} g(E, n), \tag{11}$$

where $E(t) : \Omega \to \mathbb{R}$ represents **trans-membrane voltage** at time t, $n(t) : \Omega \to \mathbb{R}$ represents a **slow current gate variable**, and D_0, τ_E, and τ_n are constant parameters. (Note: D_0 here is playing the part of $\mathbf{D}$ in the general equation. When we couple this system with the poroelastic model later, this will become a matrix depending on the stresses of that system!)

It is not important that the meaning of these variables is fully understood: mathematically, they can be considered to simply be two species that diffuse and interact. Later on we shall see that is is concentration of n that causes

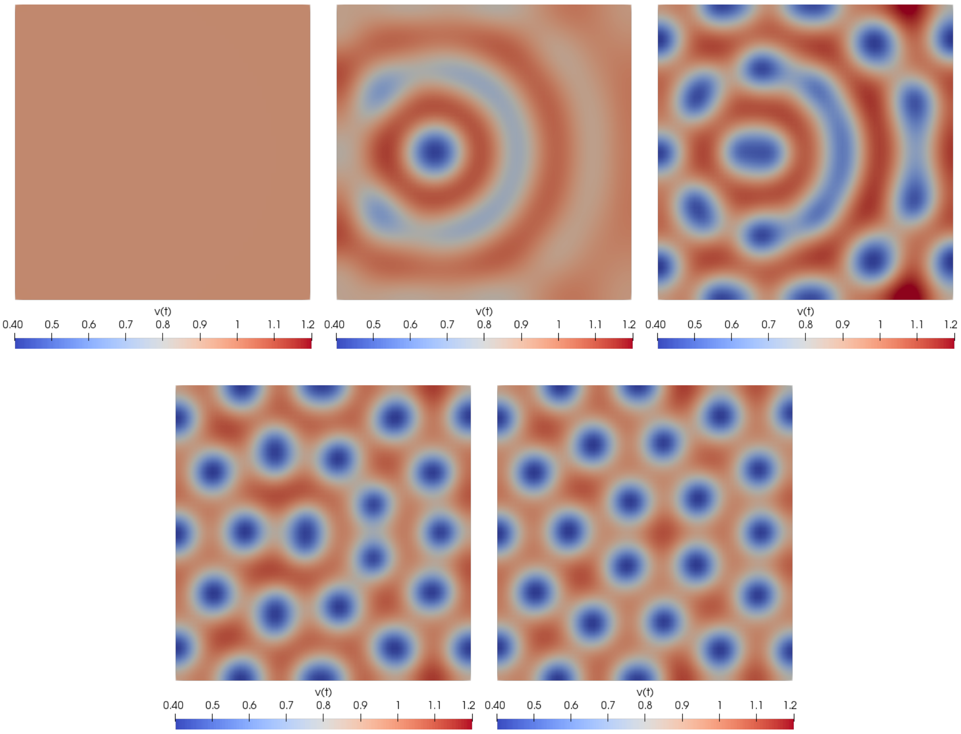

Figure 2: $v(t)$ at $t = 0, 0.244, 0.366, 0.732, 1.953$.

muscle contraction, but that can still be easily understood as a mathematical relation. For a more in-depth study of the physics, chemistry, and biology behind these equations, the reader is advised to read [2] and the works it is based on.

In the above, $f(E, n)$ and $g(E, n)$ are given by:

$$f(E, n) = -E + (E^\star - \mathcal{D}(n))h(E), \tag{12}$$

$$g(E, n) = \mathcal{R}(n)\,\theta(E - E_n) - (1 - \theta(E - E_n))n, \tag{13}$$

where $E^\star$ and E_n are constants, $\theta(x)$ is the standard **Heaviside step function**

$$\theta(x) = \begin{cases} 0 & \text{if } x \leq 0, \\ 1 & \text{if } x > 0, \end{cases}$$

and the **restitution function** $\mathcal{R}(n)$ and **dispersion function** $\mathcal{D}(n)$ represent certain electromechanical properties of the medium, and are taken in [2] to be

$$\mathcal{R}(n) = \frac{1-(1-e^{-Re})n}{1-e^{-Re}}, \tag{14}$$

$$\mathcal{D}(n) = n^M, \tag{15}$$

with Re and M once again constant parameters.

There is also a function $h(E)$ we have yet to define. This is related to steady states of polarisation in the cells of the tissue, and is given in [2] as

$$h(E) = (1-\tanh(E-E_h))\frac{E^2}{2}, \tag{16}$$

with E_h the last constant model parameter on our list.

Typical values for the model parameters—and those we will use in our numerical computation— are given in the following table:

E_h	3.0
E_n	1.0
$E^\star$	1.5415
τ_E	2.5
τ_n	250
D_0	1.1×10^3
Re	1.0
M	5.0

1.4.1 Weak form

As before, we multiply by test functions and integrate over the domain. We will multiply (10) by a test function F corresponding to E, and (11) by a test function m corresponding to n, before integrating over Ω.

$$\begin{aligned}\int_\Omega \frac{\partial}{\partial t}E\,F\,\mathrm{d}V &= D_0\int_\Omega \operatorname{div}(\nabla E)\,F\,\mathrm{d}V + \frac{1}{\tau_E}\int_\Omega f(E,n)\,F\,\mathrm{d}V,\\ \int_\Omega \frac{\partial}{\partial t}n\,m\,\mathrm{d}V &= \frac{1}{\tau_n}\int_\Omega g(E,n)\,m\,\mathrm{d}V.\end{aligned}$$

We integrate by parts the term containing $\operatorname{div}\nabla E$ to get

$$\begin{aligned}\int_\Omega \frac{\partial}{\partial t}E\,F\,\mathrm{d}V = D_0&\int_{\partial\Omega} F\operatorname{div}(\nabla E)\cdot\boldsymbol{n}\,\mathrm{d}s\\ &- D_0\int_\Omega \nabla E\cdot\nabla F\,\mathrm{d}V + \frac{1}{\tau_E}\int_\Omega f(E,n)\,F\,\mathrm{d}V,\end{aligned}$$

and if we choose F to be 0 everywhere along $\partial\Omega$ we arrive at our final weak form of the Karma equations:

$$\int_\Omega \frac{\partial}{\partial t} E\, F \,\mathrm{d}V + D_0 \int_\Omega \nabla E \cdot \nabla F \,\mathrm{d}V = \frac{1}{\tau_E} \int_\Omega f(E, n)\, F \,\mathrm{d}V, \tag{17}$$

$$\int_\Omega \frac{\partial}{\partial t} n\, m \,\mathrm{d}V = \frac{1}{\tau_n} \int_\Omega g(E, n)\, m \,\mathrm{d}V. \tag{18}$$

1.4.2 Initial conditions

Without an initial electrical stimulation, nothing will happen! We will therefore create a stimulation function to apply to the medium (which, for the purposes of the computation, we will set as $\Omega = [0, 6.72]^2$):

$$S(\boldsymbol{x}, t) = \begin{cases} S_1(\boldsymbol{x}) & \text{if } t \in [1, 4], \\ S_2(\boldsymbol{x}) & \text{if } t \in [350, 353], \\ 0 & \text{otherwise,} \end{cases}$$

where the two stimulation indicator functions S_1 and S_2 are given by

$$S_1(\boldsymbol{x}) = \begin{cases} 3 & \text{if } \boldsymbol{x} \in [0, 0.0672] \times [0, 6.72], \\ 0 & \text{otherwise,} \end{cases}$$

$$S_2(\boldsymbol{x}) = \begin{cases} 3 & \text{if } \boldsymbol{x} \in [0, 3.36]^2, \\ 0 & \text{otherwise.} \end{cases}$$

See figure 3 for a clearer view of these regions.

1.4.3 Results of computation

We present now the results of this computation. In the following diagrams, E is shown on the left, and n on the right, at the same timesteps. In figures 4–6 we see the wave move across the domain from left to right. Notice that E has a very sharp boundary between the two states, but n varies gradually, beginning to vary at the exact point that E switches rapidly. In figure 6, the wave in E has filled the entire domain, but you can see on the left-hand-side a slight decrease in concentration. At this time, n is nearing its maximum on the left. In figure 7, at $t = 300$, E has suddenly dropped to 0, as quickly as it rose. This is mirrored in n as the slow gradient begins to decrease, rather than increase.

Figure 8 is the first recorded snapshot after the second wave of stimulation is applied to the lower-left quarter of the domain. Please note that the colour scaling for n has been changed, with the lowest colour now being set to 0.5 rather than 0. Figure 9 shows the wave of low concentration moving in the

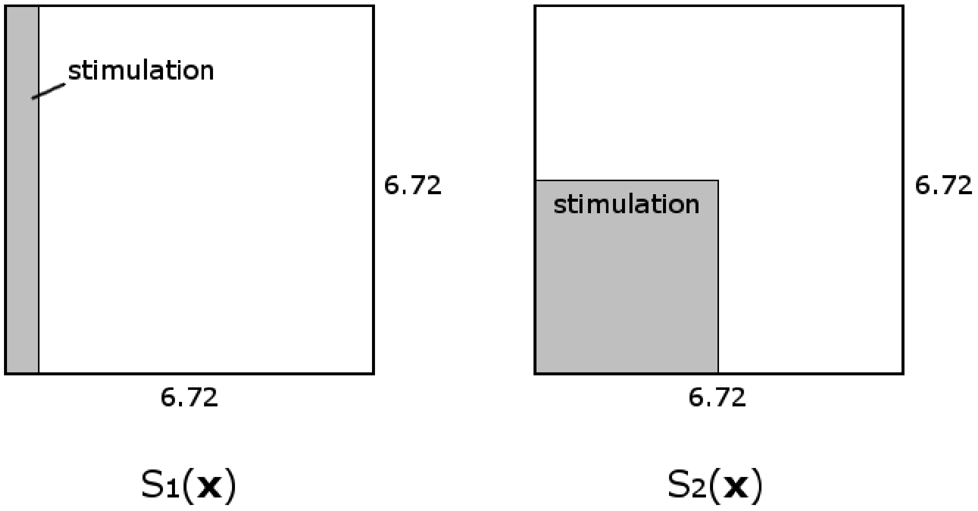

Figure 3: The areas of stimulation.

opposite direction as before, into the region to which the second stimulation was applied, with a wave of high concentration moving upwards from that region. We see a spiral pattern beginning to emerge. Finally, figures 10 and 11 show the spiral continue to grow, its shape exactly the same in E and n.

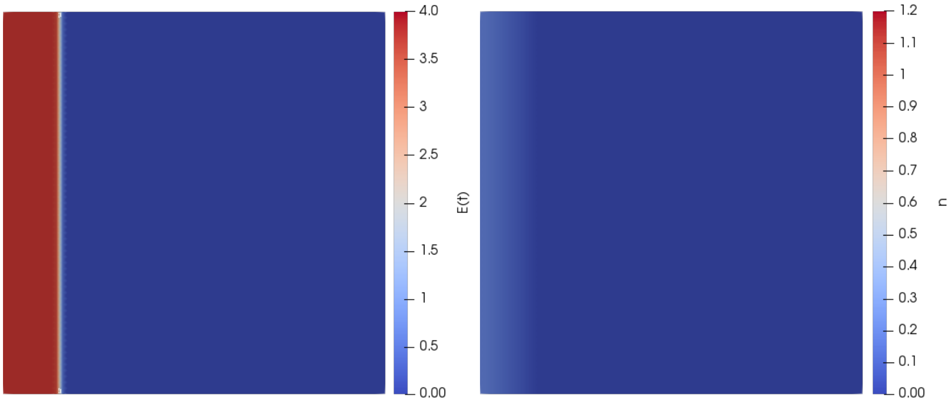

Figure 4: $t = 30$.

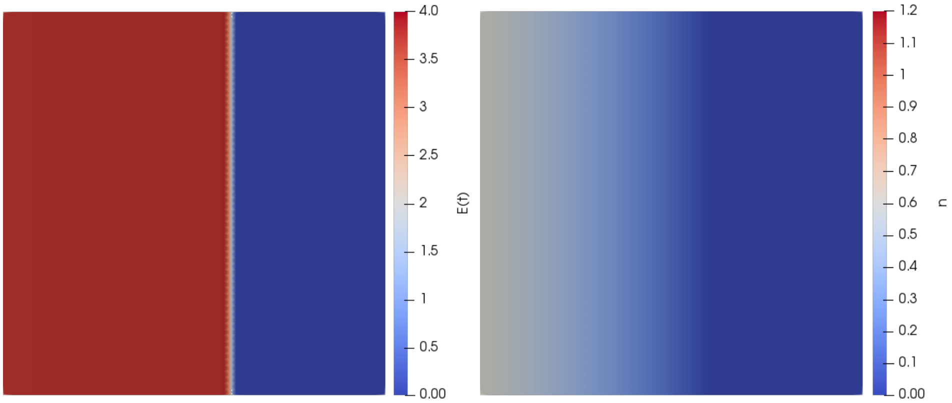

Figure 5: $t = 120$.

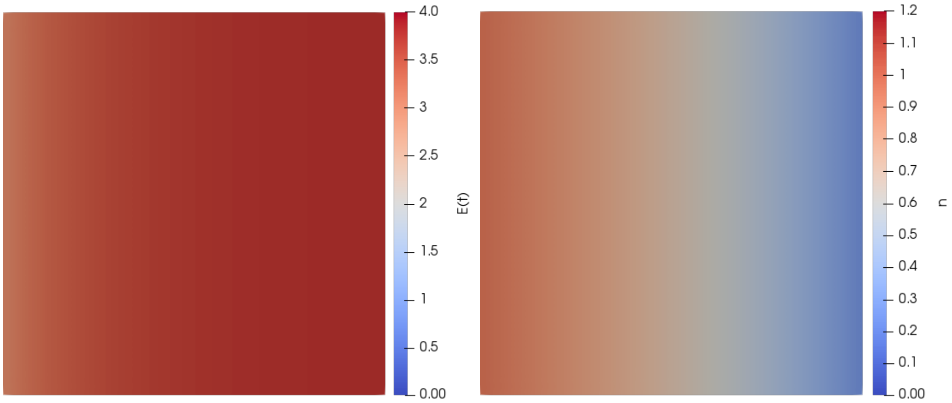

Figure 6: $t = 240$.

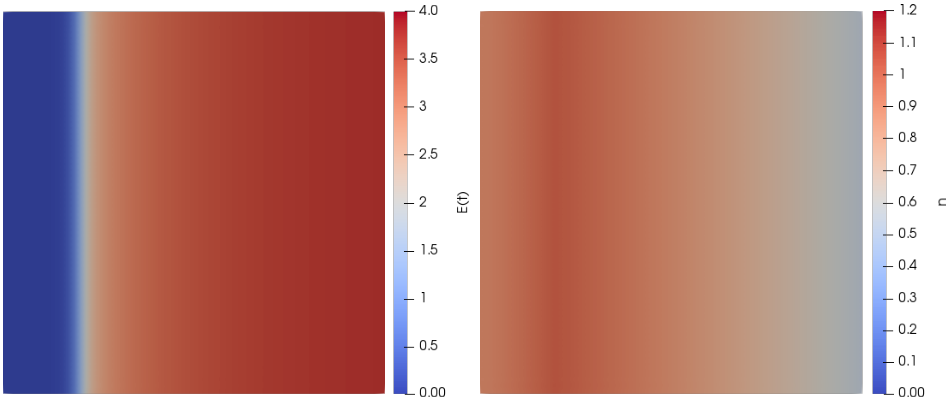

Figure 7: $t = 300$.

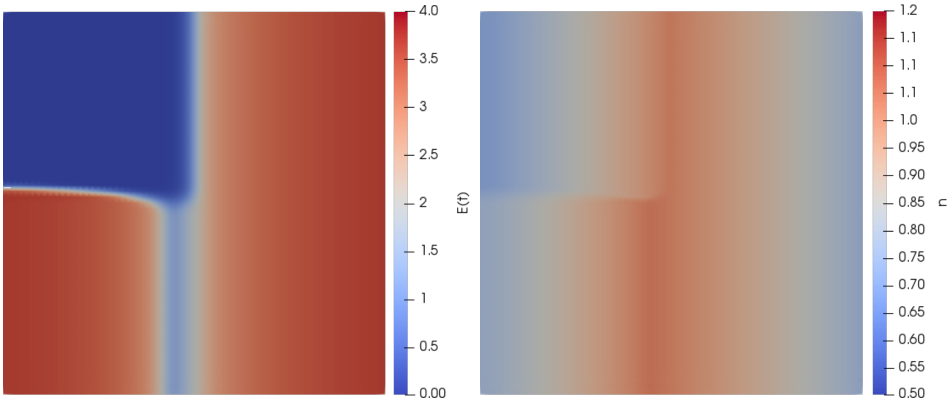

Figure 8: $t = 360$.

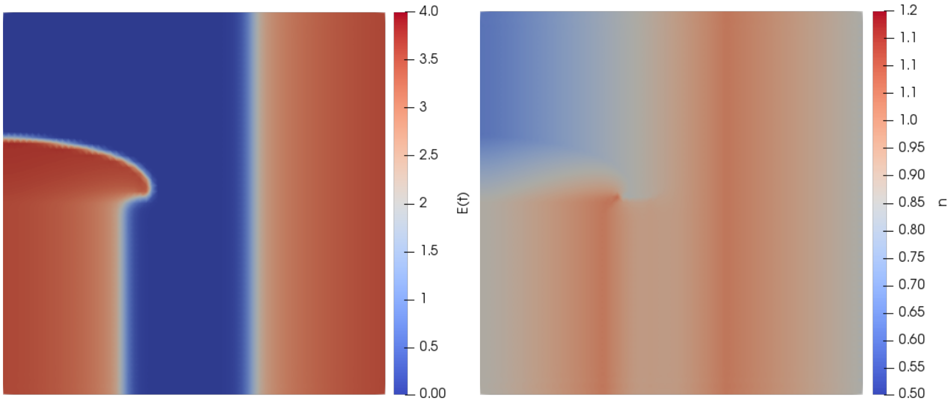

Figure 9: $t = 390$.

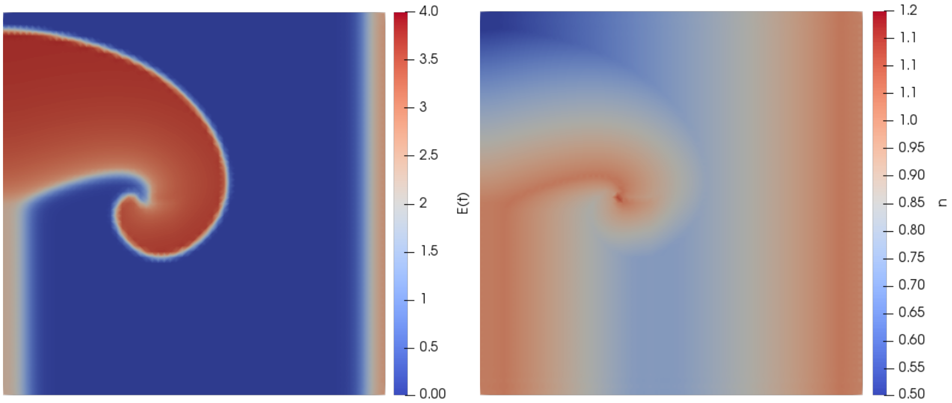

Figure 10: $t = 450$.

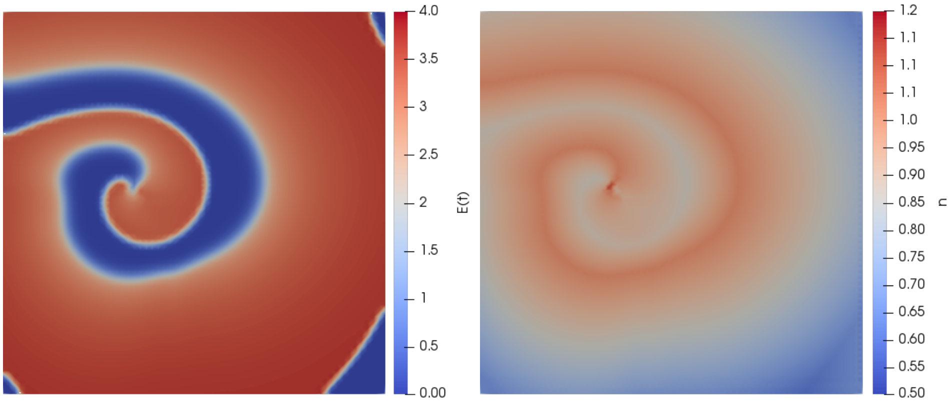

Figure 11: $t = 570$.

1.5 Conclusion

As you can see, we get very interesting results from both reaction-diffusion systems, with the Karma spirals (explained and explored more in [2]) showcasing an unexpected emergent property of some relatively simple equations. We will be taking the Karma equations forward, and combining this system with that which we are about to explore: poroelasticity.

2 Poroelasticity

The second model we are going to explore is that of **poroelasticity**. First, the system will be described, then a derivation of the model equations will be shown. Next, the finite element method will be used to numerically solve these equations for a simple set of boundary conditions, and finally, these boundary conditions will be changed, and time dependence of the system will be introduced, to show different behaviours of the system.

2.1 Introducing the model

A porous material is one that contains many small holes, through which a fluid can move. A sponge is a good example of a porous medium. Since these pores are usually very small compared to the scales we are interested in, we can *homogenise* the mixture of porous skeleton and fluid; treat it as if both skeleton and fluid are present *at each infinitesimal point.*

2.2 Derivation of equations

We will follow steps in [7] to derive a set of equations that describe a poroelastic system, and then modify these equations as in [3] to make them ready for numerical solution using the finite element method.

We are considering a poroelastic mixture, consisting of a solid porous skeleton and an incompressible pore fluid, occupying the region $\Omega(t) \subset \mathbb{R}^d$ with boundary $\partial\Omega(t)$ (where $d \in \{2, 3\}$ represents the dimensions of the system we are modelling).

In order to derive these equations, we will turn to physics, and utilise conservation laws. Unless otherwise stated, the derivations are based on those in [5], modified to suit the formulation given in [3].

2.2.1 Conservation of mass

To consider mass conservation, we must introduce some properties of the porous skeleton and fluid which will allow us to describe the amount of mass present in certain regions.

- c_0 — a positive constant known as the **constrained specific storage coefficient**, which is a measure of how much fluid can be contained inside the porous skeleton.

- α — the **Biot–Willis parameter**, a positive constant representing the response of the porous skeleton to deformation.

- $p(t)$ — the **pore pressure** of the fluid phase at a certain point. This is the first unknown that we will be solving for.
- $\boldsymbol{u}(t)$ — the **displacement** of the solid skeleton phase from its original position. This is the second unknown that we will solve for. Unlike the others, this is a *vector* function, $\boldsymbol{u}(t) : \Omega \to \mathbb{R}^d$.
- $\boldsymbol{w}(t)$ — the **specific discharge** at a point. This is another vector function, like $\boldsymbol{u}$ above, but here representing the rate of fluid crossing a unit area of the porous skeleton. In other words, it represents the motion of the fluid relative to the solid phase.
- η and ρ — the **viscosity** and **density** of the pore fluid respectively.
- κ — the **permeability** of the solid skeleton.

N.B. unless otherwise stated, these quantities are functions on space—e.g. $\kappa = \kappa(\boldsymbol{x}) = \kappa : \Omega \to \mathbb{R}$.

We now look at an arbitrary volume inside the domain, $V \subset \Omega$. The total amount of fluid inside this volume is given by the expression

$$\int_V c_0 p + \alpha(\operatorname{div}\boldsymbol{u})\,\mathrm{d}V,$$

the fluid pressure plus the material volume, as in [3]. The rate of fluid flowing out of the volume (which has boundary ∂V and outward-facing normal $\boldsymbol{n}$) is equal to

$$\int_{\partial V} \boldsymbol{w} \cdot \boldsymbol{n}\,\mathrm{d}s,$$

and we must also consider fluid entering or leaving the volume due to a source/sink term, $s(t) : \Omega \to \mathbb{R}$, which gives a rate of fluid change equal to

$$\int_V s\,\mathrm{d}V.$$

These three statements allow us to write a conservation of mass equation for the fluid,

$$\frac{\partial}{\partial t}\int_V c_0 p + \alpha(\operatorname{div}\boldsymbol{u})\,\mathrm{d}V = -\int_{\partial V} \boldsymbol{w} \cdot \boldsymbol{n}\,\mathrm{d}s + \int_V s\,\mathrm{d}V.$$

We can apply the divergence theorem to the expression in the middle, giving us

$$\frac{\partial}{\partial t}\int_V c_0 p + \alpha(\operatorname{div}\boldsymbol{u})\,\mathrm{d}V = -\int_V \operatorname{div}\boldsymbol{w}\,\mathrm{d}V + \int_V s\,\mathrm{d}V,$$

and, since V was chosen arbitrarily, we can arrive at the following:

$$\frac{\partial}{\partial t}(c_0 p + \alpha(\operatorname{div}\boldsymbol{u})) + \operatorname{div}\boldsymbol{w} = s \quad \text{in } \Omega. \tag{19}$$

This is useful, however $\boldsymbol{w}$ represents another unknown we will have to solve for. Luckily, **Darcy's Law** (a derivation of which appears in [6]) can help us here. Darcy's Law states that:

$$\boldsymbol{w} = -\mathbf{K}(\nabla p - \rho \boldsymbol{g}),$$

where $\mathbf{K}$ is the **permeability tensor**, and $\boldsymbol{g}$ is a vector representing the force of gravity. Here we will assume an isotropic skeleton phase (this means it has the same characteristics in all directions), meaning $\mathbf{K}$ can be written as $K\mathbf{I}$, with $\mathbf{I}$ the identity matrix, and $K = \kappa/\eta$ the **permeability coefficient**. We will also assume that no gravitational field is present, i.e. $\boldsymbol{g} = \mathbf{0}$.

We can substitute this into (19) to give

$$\frac{\partial}{\partial t}(c_0 p + \alpha(\operatorname{div}\boldsymbol{u})) - \frac{1}{\eta}\operatorname{div}(\kappa \nabla p) = s \quad \text{in } \Omega, \tag{20}$$

a suitable conservation of mass equation. To simplify even further, we can employ **Euler time discretisation** as in [3] to remove time-dependence and find steady state solutions, giving us the following:

$$c_0 p + \alpha(\operatorname{div}\boldsymbol{u}) - \frac{1}{\eta}\operatorname{div}(\kappa \nabla p) = s \quad \text{in } \Omega, \tag{21}$$

which we will use as our conservation of mass equation going forward.

2.2.2 Conservation of momentum

We consider again an arbitrary volume $V \subset \Omega$ with boundary ∂V and unit outward normal vector $\boldsymbol{n}$. We introduce the **Cauchy stress tensor**, denoted by $\boldsymbol{\sigma}$, where σ_{ij}, the i, j-th component of this tensor, is equal to the total force in the x_j direction acting per unit area whose normal is in the x_i direction.

The resultant of the internal forces on the boundary ∂V of our arbitrary volume is then given by

$$\int_{\partial V} \boldsymbol{\sigma}\boldsymbol{n}\, \mathrm{d}s.$$

If we assume also that a **body force** is acting per unit volume on the mixture, the total force on our arbitrary volume is given by

$$\int_V \boldsymbol{f}\, \mathrm{d}V.$$

In order to conserve momentum, these two forces must balance out:

$$\int_{\partial V} \boldsymbol{\sigma n}\,\mathrm{d}s + \int_V \boldsymbol{f}\,\mathrm{d}V = 0.$$

We can apply the divergence theorem to the integral on the left, giving us the expression

$$\int_V \mathbf{div}\,\boldsymbol{\sigma} + \boldsymbol{f}\,\mathrm{d}V = 0,$$

and since our volume V is arbitrary, we can remove the integral sign to give our final conservation of momentum equation,

$$-\mathbf{div}\,\boldsymbol{\sigma} = \boldsymbol{f} \quad \text{in } \Omega. \tag{22}$$

2.2.3 Constitutive law of the solid

A **constitutive law** is a relation between physical properties of a material, commonly related to stress-strain responses of that substance. Here, we will attempt to find an expression for $\boldsymbol{\sigma}$, the Cauchy stress tensor. To do this, we must introduce the **infinitesimal strain tensor** $\boldsymbol{\varepsilon}(\boldsymbol{u}) = \frac{1}{2}(\nabla\boldsymbol{u} + \nabla\boldsymbol{u}^T)$, which can be thought of as a "symmetrized gradient of displacements" [3], and the two **Lamé parameters** of the solid, λ and μ, which represent certain responses to stress and strain of the porous skeleton.

We will state the constitutive law as it appears in [3]:

$$\boldsymbol{\sigma} = \lambda(\operatorname{div}\boldsymbol{u})\mathbf{I} + 2\mu\boldsymbol{\varepsilon}(\boldsymbol{u}) - p\mathbf{I} \quad \text{in } \Omega. \tag{23}$$

2.2.4 Three-field strong formulation

Equations 21, 22, and 23 give us a problem in two unknowns, $\boldsymbol{u}$ and p. We introduce now, as in [3], an *auxiliary unknown* ϕ, which can be regarded as the "total pressure" of the two phases combined, and is defined by

$$\phi = p - \lambda\operatorname{div}\boldsymbol{u}. \tag{24}$$

We can now combine equations 21, 22, 23, and 24 into a system of three equations for three unknowns, $\boldsymbol{u}$, p, and ϕ:

$$\begin{aligned} \left(c_0 + \frac{\alpha}{\lambda}\right)p - \frac{\alpha}{\lambda}\phi - \frac{1}{\eta}\operatorname{div}(\kappa\nabla p) &= s && \text{in } \Omega, && (25)\\ \boldsymbol{\sigma} &= 2\mu\boldsymbol{\varepsilon}(\boldsymbol{u}) - \phi\mathbf{I} && \text{in } \Omega, && (26)\\ -\mathbf{div}\,\boldsymbol{\sigma} &= \boldsymbol{f} && \text{in } \Omega. && (27) \end{aligned}$$

2.3 Boundary conditions

We will assign boundary conditions as in [3] as follows. We assume that the boundary of the domain, $\partial\Omega$, is split disjointly into two parts, one where we impose pressure, and one where we impose displacements, respectively called Γ_p and $\Gamma_{\boldsymbol{u}}$, such that $\partial\Omega = \Gamma_p \cup \Gamma_{\boldsymbol{u}}$. For now, we will assert **homogeneous** boundary conditions, so that:

$$p = 0, \quad \boldsymbol{\sigma n} = \boldsymbol{0} \qquad \text{on } \Gamma_p, \tag{28}$$

$$\boldsymbol{u} = 0, \quad (\kappa \nabla p) \cdot \boldsymbol{n} = 0 \qquad \text{on } \Gamma_{\boldsymbol{u}}. \tag{29}$$

This nicely corresponds to $\Gamma_{\boldsymbol{u}}$ representing sections of the boundary that are static, for example held against a wall, and Γ_p representing sections of the medium that are open and can move freely. (It will be very easy to change these later on.)

2.3.1 Weak form

To allow us to numerically solve these equations using the finite element method (which will be introduced shortly), we must rewrite the equations 25–27 in **weak form**. In summary, the weak form of an equation uses multiplication by a test function and integration by parts to move derivatives from our unknowns (here $\boldsymbol{u}$, p and ϕ) onto the test functions, allowing us to find non-differentiable functions that satisfy the equations.

First, let us multiply (25) by a test function q (corresponding to pressure p), and integrate over Ω:

$$\left(c_0 + \frac{\alpha}{\lambda}\right) \int_\Omega pq \,\mathrm{d}V - \frac{\alpha}{\lambda} \int_\Omega \phi q \,\mathrm{d}V - \frac{1}{\eta} \int_\Omega \operatorname{div}(\kappa \nabla p)\, q \,\mathrm{d}V = \int_\Omega sq \,\mathrm{d}V.$$

We integrate by parts the term containing $1/\eta$, giving

$$\begin{aligned}\left(c_0 + \frac{\alpha}{\lambda}\right) \int_\Omega pq \,\mathrm{d}V - \frac{\alpha}{\lambda} \int_\Omega \phi q \,\mathrm{d}V& \\ + \frac{1}{\eta}\left(\int_\Omega \kappa \nabla p \cdot \nabla q \,\mathrm{d}V - \int_{\partial\Omega} \kappa \nabla p \cdot \boldsymbol{n}\, q \,\mathrm{d}s\right) &= \int_\Omega sq \,\mathrm{d}V.\end{aligned}$$

Since

$$\int_{\partial\Omega} \kappa \nabla p \cdot \boldsymbol{n}\, q \,\mathrm{d}s = \int_{\Gamma_p} \kappa \nabla p \cdot \boldsymbol{n}\, q \,\mathrm{d}s + \int_{\Gamma_{\boldsymbol{u}}} \kappa \nabla p \cdot \boldsymbol{n}\, q \,\mathrm{d}s,$$

we will be able to remove this term. Recall the boundary condition (29), which allows us to remove the integral over $\Gamma_{\boldsymbol{u}}$. To remove the other part, we simply assert that $q = 0$ on Γ_p!

We remove this term and divide by α to give our first weak form equation:

$$\left(\frac{c_0}{\alpha}+\frac{1}{\lambda}\right)\int_\Omega pq\,\mathrm{d}V-\frac{1}{\lambda}\int_\Omega \phi q\,\mathrm{d}V \\ +\frac{1}{\eta\alpha}\int_\Omega \kappa\nabla p\cdot\nabla q\,\mathrm{d}V=\frac{1}{\alpha}\int_\Omega sq\,\mathrm{d}V \qquad (30)$$

for all q such that $q=0$ on Γ_p.

We perform similar steps and introduce test functions ψ and $\boldsymbol{v}$ corresponding respectively to ϕ and $\boldsymbol{u}$ to give our other two weak form equations:

$$-\frac{1}{\lambda}\int_\Omega \phi\psi\,\mathrm{d}V+\frac{1}{\lambda}\int_\Omega p\psi\,\mathrm{d}V-\int_\Omega \psi\mathrm{div}\,\boldsymbol{u}\,\mathrm{d}V=0 \quad \forall\psi, \qquad (31)$$

$$2\mu\int_\Omega \boldsymbol{\varepsilon}(\boldsymbol{u}):\boldsymbol{\varepsilon}(\boldsymbol{v})\,\mathrm{d}V-\int_\Omega \phi\mathrm{div}\,\boldsymbol{v}\,\mathrm{d}V=\int_\Omega \boldsymbol{f}\cdot\boldsymbol{v}\,\mathrm{d}V \qquad (32)$$

$$\forall\boldsymbol{v}\text{ s.t. }\boldsymbol{v}=\boldsymbol{0}\text{ on }\Gamma_{\boldsymbol{u}},$$

where $\mathbf{A}:\mathbf{B}=\mathrm{trace}\,(\mathbf{A}^T\mathbf{B})$.

We are now ready to use the finite element method to numerically solve this system.

2.4 Initial computations

For our test problem, we will use the 'footing' problem used as an example in [3]. Our domain Ω becomes the 2-dimensional space $[-50,50]\times[0,75]$. We impose zero displacement on the lower three walls of the system, as if we have a sponge resting in a tin with high sides (although we are neglecting the force of gravity). On the upper surface we impose zero pressure, except for a region (the 'foot') where we apply a downwards force:

$$\boldsymbol{\sigma n}=\begin{pmatrix}0\\-\sigma_0\end{pmatrix}\quad\text{on this region.}$$

We will call the free region Γ_p^1, and the region over which the foot acts Γ_p^2. For us to be able to do this (as this is a Neumann, not a Dirichlet, boundary condition) it must be 'baked into' the weak form of the equation, adding a term to (32):

$$2\mu\int_\Omega \boldsymbol{\varepsilon}(\boldsymbol{u}):\boldsymbol{\varepsilon}(\boldsymbol{v})\,\mathrm{d}V-\int_\Omega \phi\mathrm{div}\,\boldsymbol{v}\,\mathrm{d}V \\ =\int_\Omega \boldsymbol{f}\cdot\boldsymbol{v}\,\mathrm{d}V+\int_{\Gamma_p^2}\begin{pmatrix}0\\-\sigma_0\end{pmatrix}\cdot\boldsymbol{v}\,\mathrm{d}V.$$

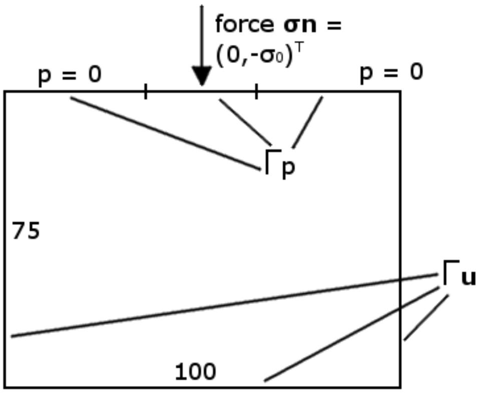

Figure 12: The 'footing' problem.

See figure 12 for a diagram of this setup.

We will subdivide the domain into 52 cells from left to right, and 39 up and down. These choices of numbers may seem arbitrary, but they allow the cells to be square in shape.

Figures 13 and 14 show the results of this computation, where the mesh has been deformed by the values of $\boldsymbol{u}$. Note the clear swelling up of the medium around the foot, due to the incompressible fluid being moved out of the way by the pressure, and also the clear increase in pressure directly underneath the foot, spreading throughout the medium. As an aside, showing a wireframe representation of the medium (as in figure 15) allows us to see the displacements within, granting us extra insight—you can see the spongy material being forced outwards underneath the foot, and spreading out near the bulges to either side.

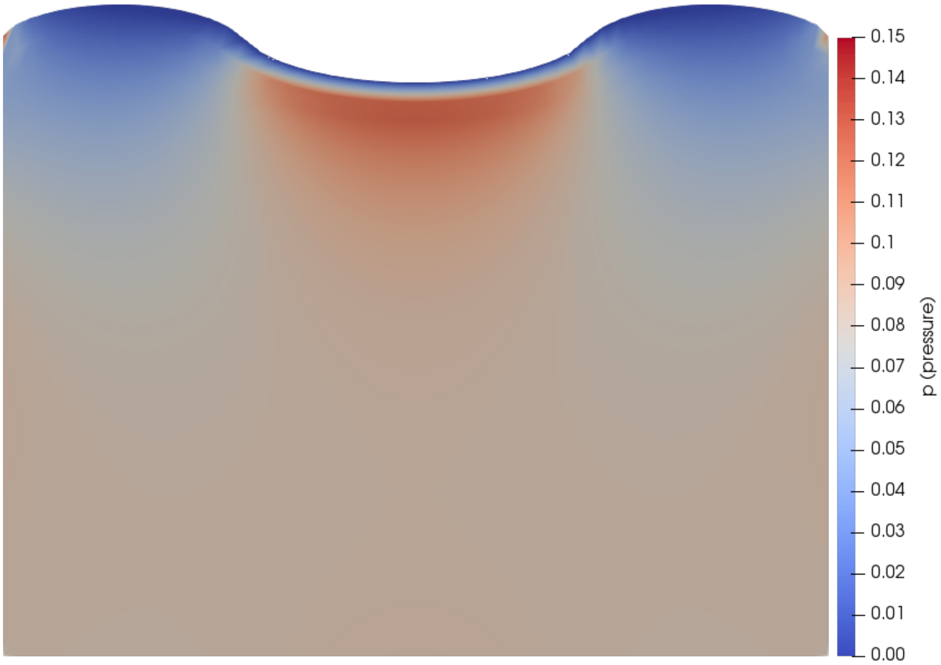

Figure 13: p in the stationary 'footing' problem.

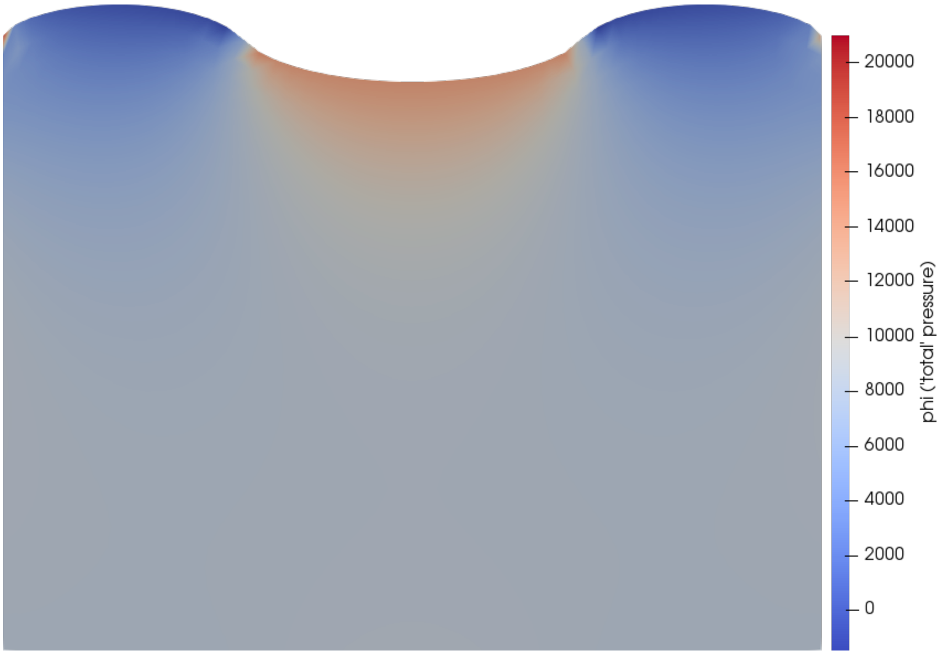

Figure 14: ϕ in the stationary 'footing' problem.

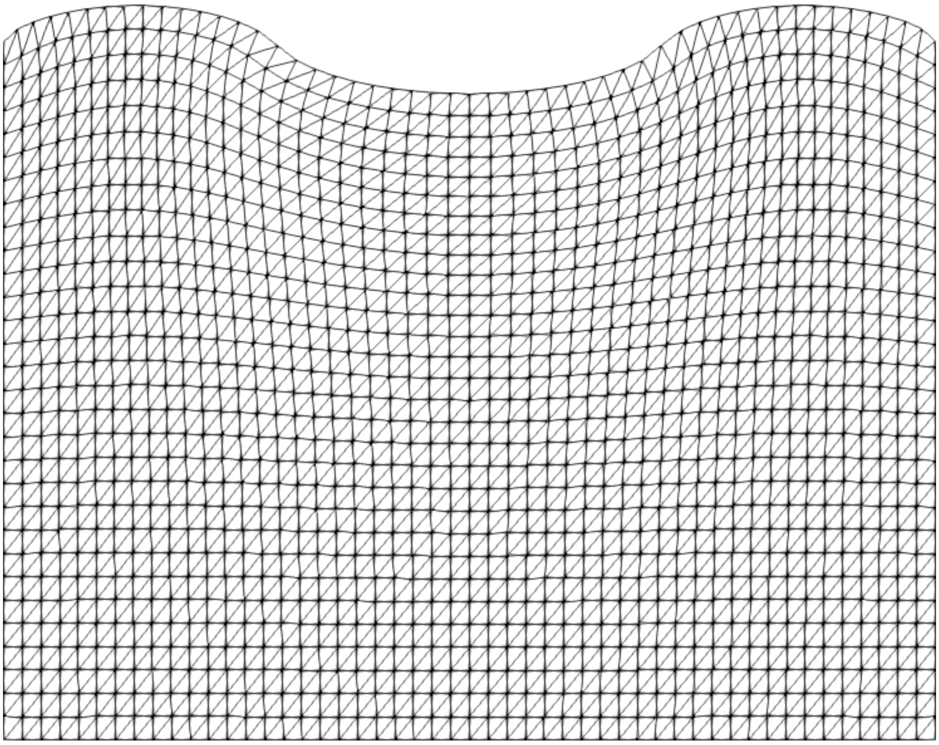

Figure 15: A wireframe rendering of the medium.

2.5 Further computations

Since the weak form we described before is general, we only need edit the boundary conditions to create an entirely new system.

2.5.1 Removing walls

We can change the system substantially by moving sections of the boundary from $\Gamma_{\boldsymbol{u}}$ to Γ_p, which physically translates to 'removing a wall' from the 'box' we have put the sponge in. We will not spend too much time on the specifics, rather only presenting the results.

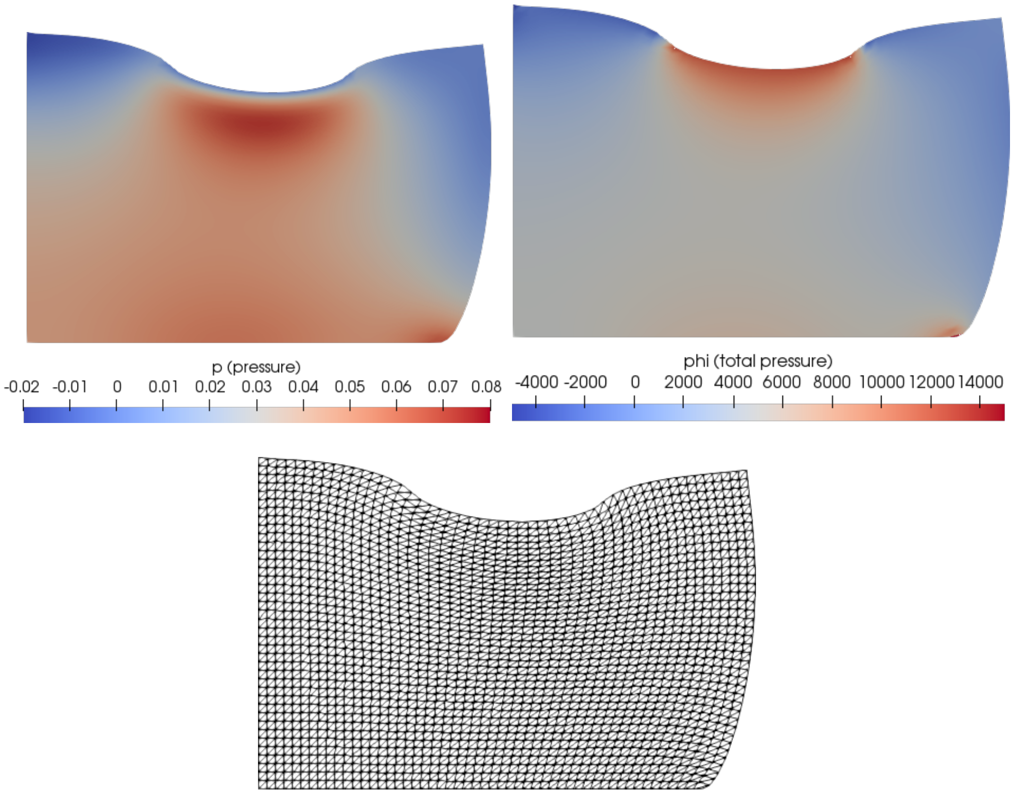

Figure 16: p, ϕ, and a wireframe rendering of the medium with one wall removed.

In figure 16 we have removed the wall on the right (i.e. prescribed $p = 0$ and not $\boldsymbol{u} = \boldsymbol{0}$ on it). You can clearly see the sponge 'seeping out', and a reduced pressure from the case with three walls is visible. The wireframe rendering shows much less 'bunching up' of the points, which you would expect to see as the sponge has more freedom to move away from the foot. Interestingly, there is an increase in pressure near the corner where the lower wall meets the free surface, which is most likely an artifact of the computation—perhaps a conflict between the two different boundary conditions, which are *both* defined at these points. (The same artifact is visible to a lesser degree on the previous diagrams.)

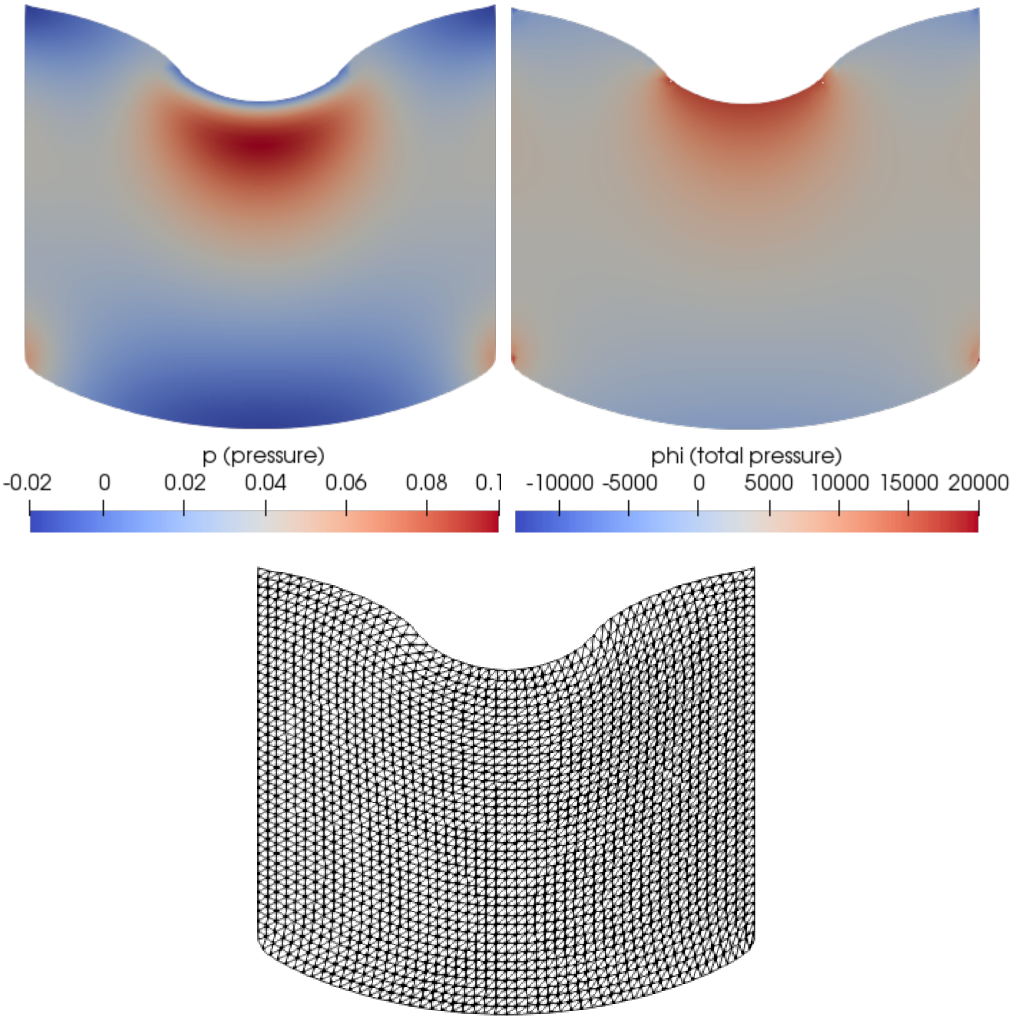

Figure 17: p, ϕ, and a wireframe rendering of the medium with the floor removed.

For figure 17, the lower surface (the ‘floor’) has been removed. The result is as we’d expect: the whole medium bulges outwards from the bottom. (Of course, in reality we would expect the sponge to be ejected from below, but we prescribed completely zero displacement on the walls, rather than just zero *normal* displacement—$\boldsymbol{u} = \boldsymbol{0}$ rather than $\boldsymbol{u} \cdot \boldsymbol{n} = 0$—so the sponge is completely fixed to the walls, and cannot slide across them.) The wireframe again shows more freedom of movement in the sponge.

2.5.2 3 dimensions

Nowhere in the derivation of this model did we specify a dimension, so the exact same equations apply to 3 dimensions as well as 2. We generalise the footing problem to 3D, swapping the foot for a circular region on the upper face, and changing the dimensions to those of a cube. Figures 18 and 19 show in detail the results of the computation. You can see, by the fact that the cutaway shows mostly low pressure, that adding the third dimension has made the medium 'stiffer' somewhat, as the force has had to be increased by a factor of 400 to get a visible displacement.

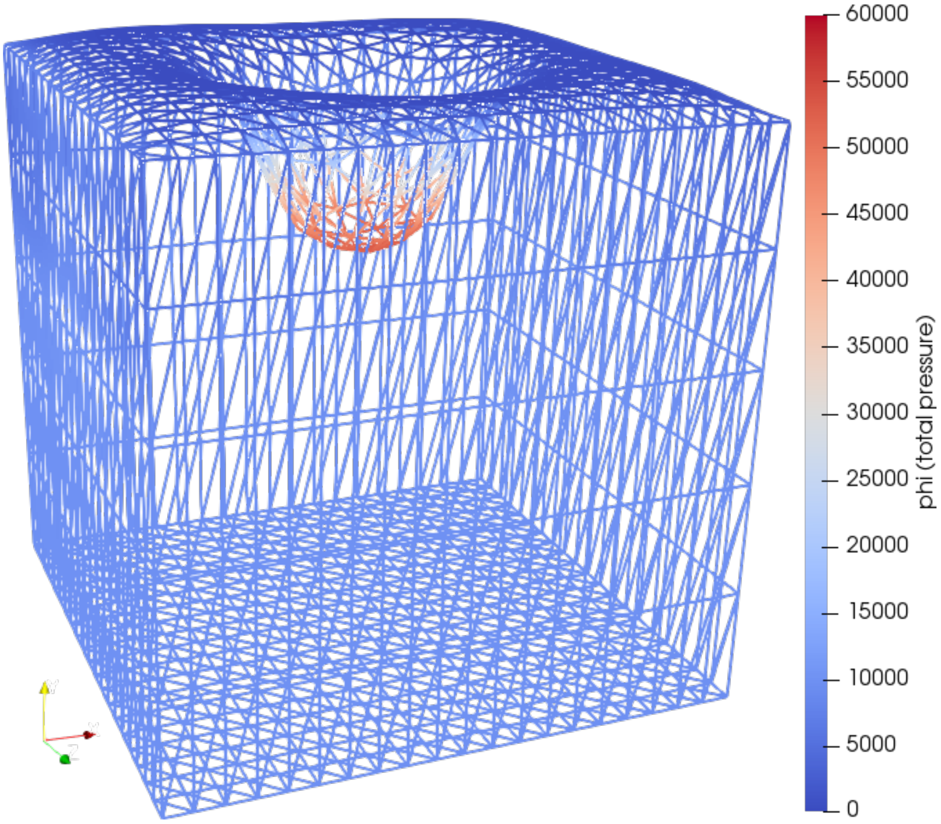

Figure 18: ϕ in the 3D footing problem. Rendered in wireframe, to allow the area under the foot to be clearly seen.

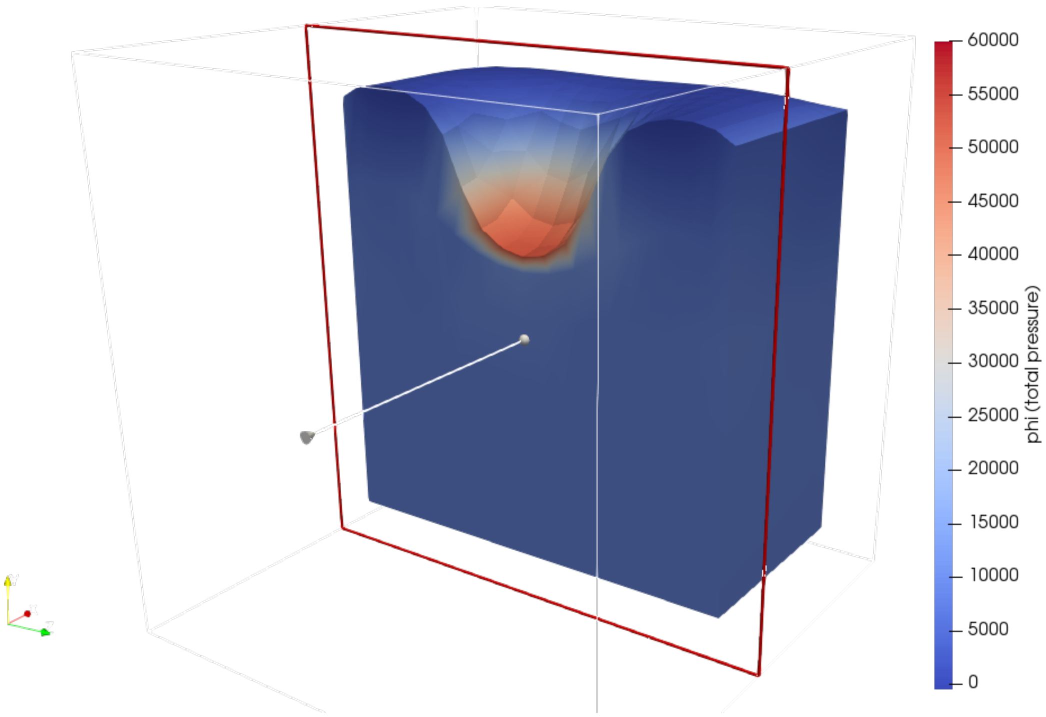

Figure 19: ϕ in the 3D footing problem. A cutaway is shown, to allow the inside to be seen in more detail.

2.6 Time-dependence

Since we will later be combining this model with reaction-diffusion, which is time-dependent, we must modify this system to be time-dependent also. Recall equation 20:

$$\frac{\partial}{\partial t}(c_0 p + \alpha(\operatorname{div} \boldsymbol{u})) - \frac{1}{\eta}\operatorname{div}(\kappa \nabla p) = s \quad \text{in } \Omega.$$

Previously, we removed this time-dependence, but now we must include it. First, we must substitute in ϕ, our auxiliary variable (and divide by α):

$$\left(\frac{c_0}{\alpha} + \frac{1}{\lambda}\right)\frac{\partial}{\partial t}p - \frac{1}{\lambda}\frac{\partial}{\partial t}\phi - \frac{1}{\eta\alpha}\operatorname{div}(\kappa \nabla p) = \frac{s}{\alpha} \quad \text{in } \Omega.$$

We will use the same time-discretisation that we used for reaction-diffusion:

$$\frac{\partial}{\partial t}p(t^n) \approx \frac{p^{n+1} - p^n}{\Delta t},$$
$$\frac{\partial}{\partial t}\phi(t^n) \approx \frac{\phi^{n+1} - \phi^n}{\Delta t},$$

giving us the time-discretised strong form of this model:

$$\begin{aligned}
\left(\frac{c_0}{\alpha}+\frac{1}{\lambda}\right)\frac{p^{n+1}}{\Delta t}-\frac{1}{\lambda}\frac{\phi^{n+1}}{\Delta t}-\frac{1}{\eta\alpha}\mathrm{div}\left(\kappa\nabla p^{n+1}\right) &= \frac{s}{\alpha} \\
+\left(\frac{c_0}{\alpha}+\frac{1}{\lambda}\right)\frac{p^{n}}{\Delta t}-\frac{1}{\lambda}\frac{\phi^{n}}{\Delta t} & \qquad\qquad \text{in } \Omega, \\
\phi^{n+1}-p^{n+1}+\lambda\,\mathrm{div}\left(\boldsymbol{u}^{n+1}\right) &= 0 \qquad\qquad \text{in } \Omega, \\
-\mathbf{div}\left(2\mu\boldsymbol{\varepsilon}\left(\boldsymbol{u}^{n+1}\right)-\phi^{n+1}\mathbf{I}\right) &= \boldsymbol{f} \qquad\qquad \text{in } \Omega,
\end{aligned}$$

and the time-discretised weak form,

$$\begin{aligned}
\left(\frac{c_0}{\alpha}+\frac{1}{\lambda}\right)\int_\Omega \frac{p^{n+1}}{\Delta t}q\,\mathrm{d}V-\frac{1}{\lambda}\int_\Omega \frac{\phi^{n+1}}{\Delta t}q\,\mathrm{d}V+\frac{1}{\eta\alpha}\int_\Omega \kappa\nabla p\cdot\nabla q\,\mathrm{d}V \\
=\frac{1}{\alpha}\int_\Omega sq\,\mathrm{d}V+\left(\frac{c_0}{\alpha}+\frac{1}{\lambda}\right)\int_\Omega \frac{p^{n}}{\Delta t}q\,\mathrm{d}V-\frac{1}{\lambda}\int_\Omega \frac{\phi^{n}}{\Delta t}q\,\mathrm{d}V \quad \forall q,
\end{aligned}$$

$$\begin{aligned}
-\frac{1}{\lambda}\int_\Omega \phi^{n+1}\psi\,\mathrm{d}V+\frac{1}{\lambda}\int_\Omega p^{n+1}\psi\,\mathrm{d}V-\int_\Omega \psi\mathrm{div}\,\boldsymbol{u}^{n+1}\,\mathrm{d}V=0 \quad \forall\psi, \\
2\mu\int_\Omega \boldsymbol{\varepsilon}\left(\boldsymbol{u}^{n+1}\right):\boldsymbol{\varepsilon}(\boldsymbol{v})\,\mathrm{d}V-\int_\Omega \phi^{n+1}\mathrm{div}\,\boldsymbol{v}\,\mathrm{d}V=\int_\Omega \boldsymbol{f}\cdot\boldsymbol{v}\,\mathrm{d}V \\
\forall\boldsymbol{v} \quad \text{s.t.} \quad \boldsymbol{v}=\mathbf{0} \text{ on } \Gamma_{\boldsymbol{u}}.
\end{aligned}$$

2.6.1 Computation

Initial computation with a constant force proved very uninteresting, as the medium reaches a stationary state in the first timestep (the speed of sound in an incompressible fluid is very fast!) so we propose an increasing force:

$$\boldsymbol{\sigma n}=\begin{pmatrix} 0 \\ -\sigma_0 t \end{pmatrix},$$

over ten timesteps of size 0.1 each.

Figures 20 and 21 clearly show an increasing pressure with the same general pattern as we saw before, and figure 22 shows in detail the bulging out over time we would expect to see in this situation.

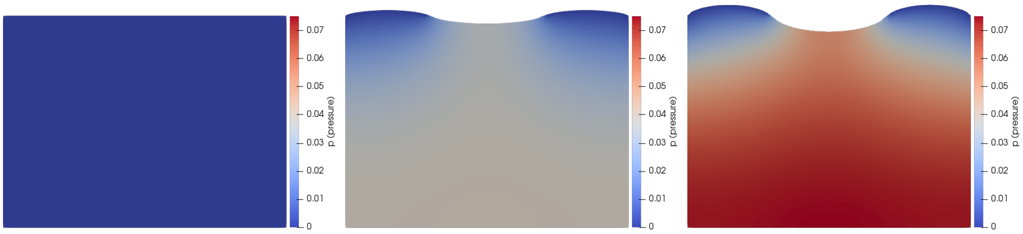

Figure 20: p at $t = 0.0, 0.5, 1.0$.

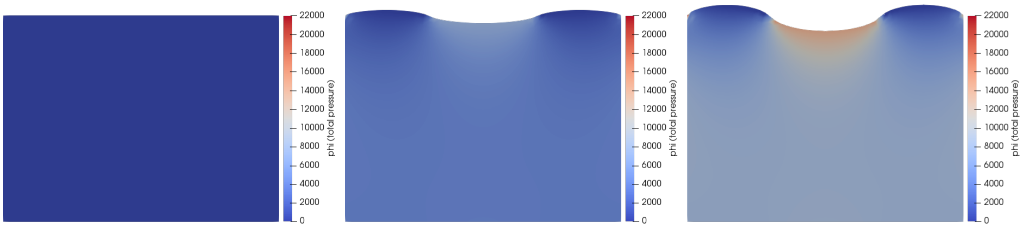

Figure 21: ϕ at $t = 0.0, 0.5, 1.0$.

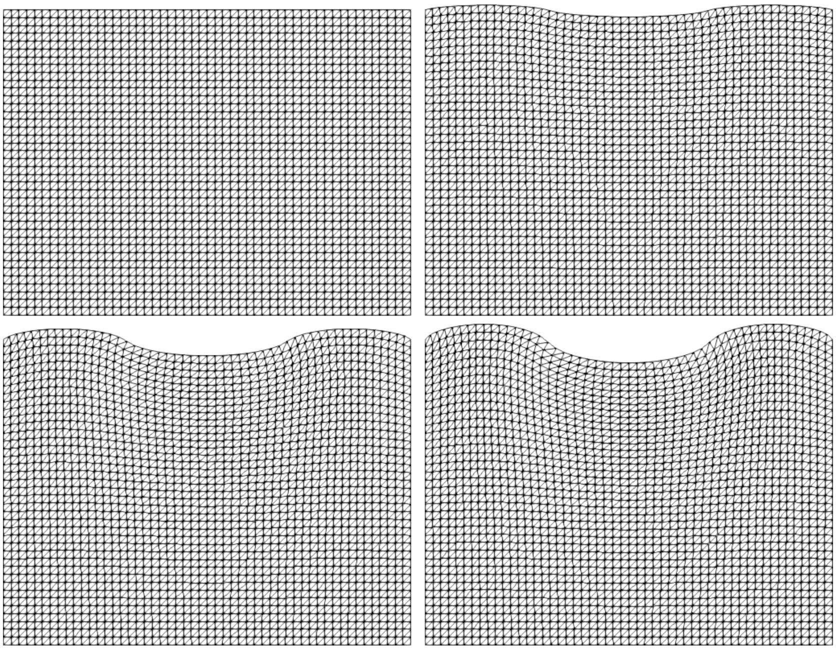

Figure 22: Wireframe rendering at $t = 0.0, 0.3, 0.7, 1.0$.

2.7 Conclusion

The numerical solutions of this problem make physical sense, and respond to the boundary conditions as we would expect. The ability to add body forces and internal stresses to the material make this model perfectly suited to being modified with the reaction-diffusion equations, as we will do in the final section of this project.

3 Combining the models

In this section, we will combine the two models we have discussed so far into a single model (anisotropic stress-assisted reaction-diffusion in a poroelastic medium), and use the finite element method to numerically solve that model for specific conditions.

This section constitutes original work, motivated and assisted in part by this project's supervisor, Ricardo Ruiz-Baier[2].

3.1 Motivation

Poroelastic systems exist everywhere. Your kitchen sponge, the cushioning in the seats of your car, and even the human brain are all examples of such media. Frequently, too, we would wish to know the exact properties of species diffusing through these media: ink making its way through a sponge, for example, or dye used in medical scans moving through a patient's brain tissue. While simple models for such a system have been postulated, none of them take anisotropic (that is, different in every direction) stresses into account. As such, this is a desirable model to create and perfect.

3.2 Adaptation of models

Recall equation 10 from the Karma reaction-diffusion model:

$$\frac{\partial}{\partial t}E = D_0 \operatorname{div}(\nabla E) + \frac{1}{\tau_E} f(E, n).$$

As pointed out before, note the constant D_0 playing the role of the diffusion tensor $\mathbf{D}$. It is here that we will modify this system with terms from the poroelasticity model.

We propose replacing D_0 with a tensor function:

$$\mathbf{D}(E, \boldsymbol{\sigma}) = D_0(1 + D_0 E)\mathbf{I} + D_0^2 \boldsymbol{\sigma}, \tag{33}$$

[2] ruizbaier@maths.ox.ac.uk

with E and D_0 remaining the same as before, $\mathbf{I}$ the identity matrix, and $\boldsymbol{\sigma}$ the stress tensor taken from the poroelasticity model.

We must, however, also modify the stress tensor. We are modelling this new medium as one that compresses when electrically stimulated, as if it is a muscle, and so presence of the current gate variable n actually causes forces to appear in the medium. Muscles have long, thin cells, all aligned in the same direction, and it is along this direction that those cells contract, and thus exert a force.

We propose that $\boldsymbol{\sigma}_{\text{total}}$, this 'total' stress, is created by combining the stress from the poroelastic model defined in (23), which we will redefine as $\boldsymbol{\sigma}_{\text{PE}}$, and the forces added by the muscle contraction, like so:

$$\boldsymbol{\sigma}_{\text{total}} = \boldsymbol{\sigma}_{\text{PE}} + c_1 n(\boldsymbol{f}_0 \otimes \boldsymbol{f}_0), \tag{34}$$

where c_1 is a **force scaling constant**, used to scale the magnitude of the force caused by the muscle contraction, n is the gate current variable from the Karma equations, and $\boldsymbol{f}_0$ is a unit vector representing the direction in which the muscle fibres run. The symbol $\otimes$ represents the *outer product* of two vectors, defined by

$$\begin{aligned} \boldsymbol{u} \otimes \boldsymbol{v} &= \mathbf{w} \quad \text{a matrix, with} \\ w_{ij} &= u_i v_j. \end{aligned}$$

We will take $\boldsymbol{f}_0 = (0, 1)^T$ initially, such that the muscle fibres run lengthways across the medium, and so

$$\boldsymbol{f}_0 \otimes \boldsymbol{f}_0 = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \otimes \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}.$$

We must also add a term into the reaction-diffusion equations to take into account the movement of the medium, thanks to Reynolds' Transport Theorem. All this does is add a term

$$\ldots + \int_\Omega (\boldsymbol{z} \cdot \nabla c) d \, \mathrm{d}V$$

into the weak formulation for each species and its test function, with c and d being the concentration and corresponding test function, and $\boldsymbol{z}$ being the velocity of the medium. We would therefore be adding

$$\ldots + \int_\Omega (\boldsymbol{z} \cdot \nabla E) F \, \mathrm{d}V + \int_\Omega (\boldsymbol{z} \cdot \nabla n) m \, \mathrm{d}V$$

to the left-hand side of the Karma weak formulation.

3.3 Computation setup

In the actual computation, we will use the same time discretisation we have used thus far, and perform the solution of the poroelastic and reaction-diffusion systems sequentially. Our algorithm looks like this:

1. Define $\boldsymbol{u}^0$, ϕ^0, p^0, E^0, and n^0, the initial values of our variables. (We will set them all to 0 everywhere.) Set $t = 0$ and $n = 0$.[3]

2. While $t < T$, our final time:

   (a) Use the poroelasticity equations, using the values ϕ^n, p^n, and n^n, to solve the system for $\boldsymbol{u}^{n+1}$, ϕ^{n+1}, and p^{n+1}. (We do not need $\boldsymbol{u}^n$, as it does not appear in the required equations.)

   (b) Find the new value of $\boldsymbol{\sigma}^n_{\text{total}}$ from ϕ^{n+1}, p^{n+1}, and n^n (as we don't know n^{n+1} yet), and calculate an approximation for the medium velocity $\boldsymbol{z}^n$ as $\left(\boldsymbol{u}^{n+1} - \boldsymbol{u}^n\right)/\Delta t$.

   (c) Solve the reaction-diffusion problem with the Karma equations, finding E^{n+1} and n^{n+1} from E^n, n^n, $\boldsymbol{z}^n$, and $\boldsymbol{\sigma}^n_{\text{total}}$.

   (d) Increase n by 1, and t by Δt.[4]

For our boundary conditions, we will fix the bottom and left walls at $\boldsymbol{u} = \boldsymbol{0}$, and allow the other two walls to be free, $p = 0$, and, through trial and error, we have set $c_1 = 1 \times 10^5$.

3.4 Computation results

We start with a timeline of snapshots of E over time in figure 23. As you can clearly see, the wave of electricity causes the medium to bunch up very slightly, before relaxing. However, as soon as the spiral takes hold, the medium contorts quite drastically, with far larger magnitudes of displacement than we saw previously. You can see also how the diffusion of the signal through the medium is itself changed by the stresses it experiences.

One more interesting thing to note is the pressure waves, visible in figure 24, that follow this spiral pattern as the medium contorts.

[3] Hopefully it is clear from context when we mean n the current gate variable, and when we mean n the current value of the timestep.

[4] Here n is the timestep counter.

3.5 Next steps

Taking this model further would involve more computation: changing the boundary conditions, introducing a periodic stimulation to simulate the pacemakers in heart tissue, or even using different reaction-diffusion equations instead of the Karma equations to show ink diffusing through a sponge, or a similar such system. This model is very extensible.

4 Conclusion

In this project, we introduced two separate models, ran numerical simulations of the systems they represent using the equations we derived, and finally brought these two models together into one, coupling them together in two different directions, into one highly useful and extensible system of equations that could be used to model and simulate physical systems as complicated as radioactive ink through the human brain, and as benign as the sponge on your kitchen sink.

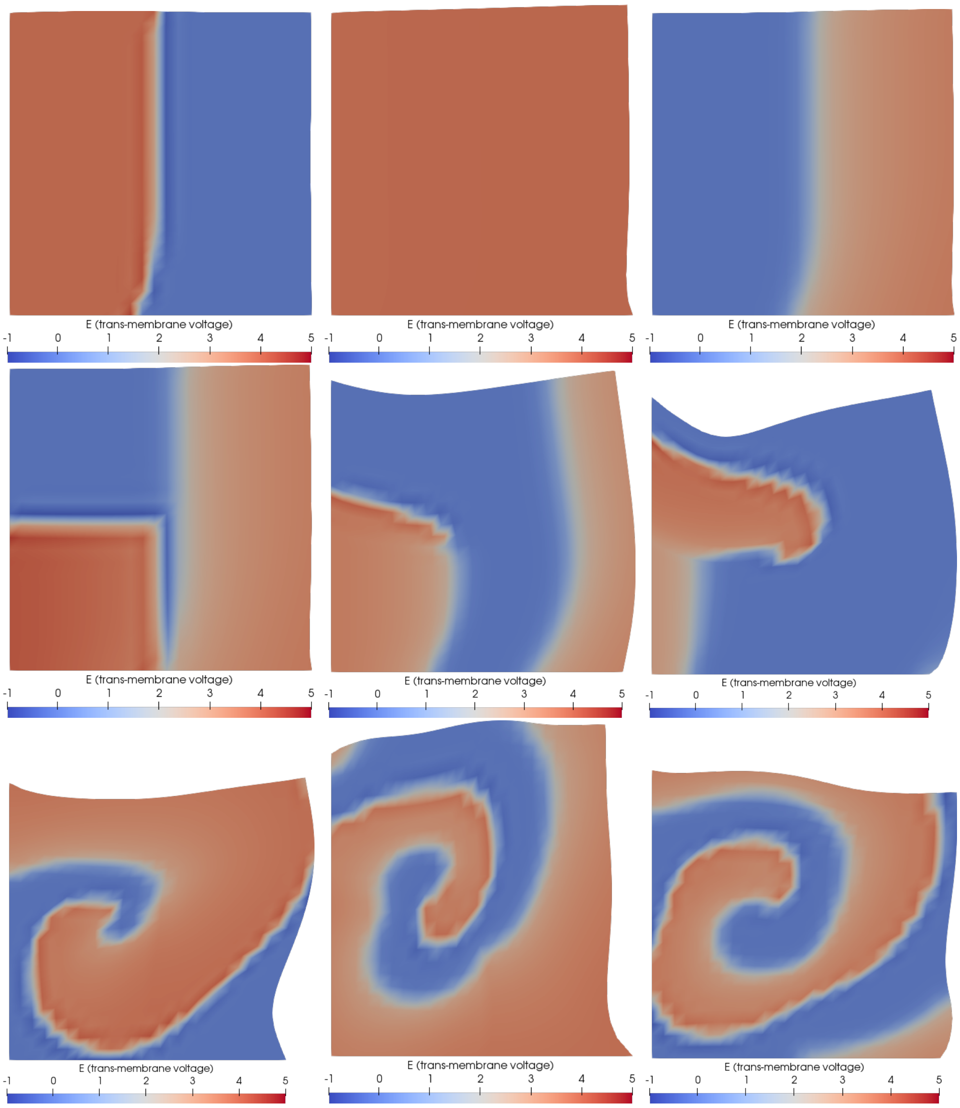

Figure 23: Values of E at t =54, 135, 318, 321, 345, 381, 453, 522, and 597.

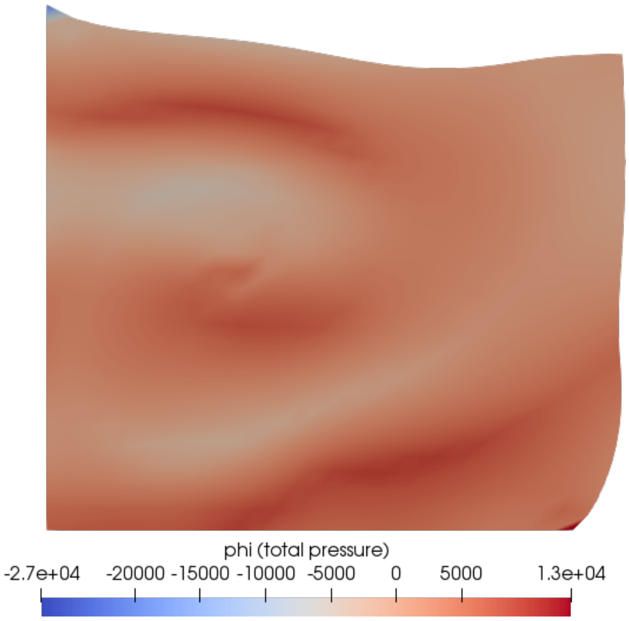

Figure 24: ϕ at $t = 579$.

Appendices

A The finite element method

The following is a very brief overview of the major idea behind the finite element method, without getting too rigorous or in-depth. It is presented simply in order for the reader to understand what the method entails, rather than instruct the reader how precisely to carry the method out themselves. For a far more thorough, yet remarkably easy to understand, introduction to this method, turn to [4]. This subsection is loosely adapted from that work.

A.1 Example problem

The problem we intend to solve is the following:

$$\begin{aligned} -\nabla^2 u + 2c &= f && \text{in } \Omega, \\ u &= g_0 && \text{on } \Gamma_D, \\ \nabla u \cdot \boldsymbol{n} &= g_1 && \text{on } \Gamma_N, \end{aligned}$$

where we are trying to find the values of a function u over a spatial domain Ω. We will assume initially that this boundary is two-dimensional, and has a piecewise linear boundary (i.e. a boundary made up of straight lines) on which we impose disjoint Dirichlet and Neumann boundary conditions, $\partial\Omega = \Gamma_D \cup \Gamma_N$, with $\boldsymbol{n}$ an outward-facing unit normal.

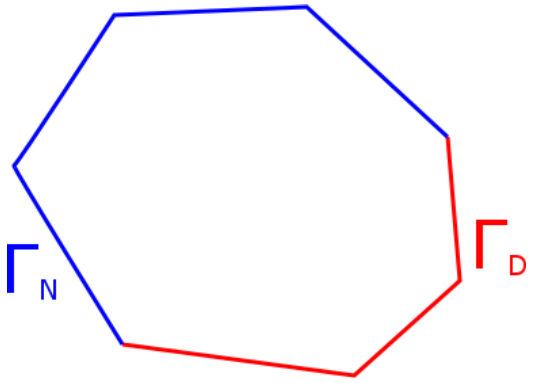

Figure 25: The domain, Ω, with boundary $\partial\Omega = \Gamma_D \cup \Gamma_N$.

A.2 Preparation of domain

We will **triangulate** the domain, which means we will divide it up into triangles. These triangles must only meet at vertices or edges, they may not have e.g. a vertex of one triangle lying on an edge of another. Let N be the number of vertices created by such a triangulation, and let $\boldsymbol{p}_i$ be the i-th vertex (numbered arbitrarily).

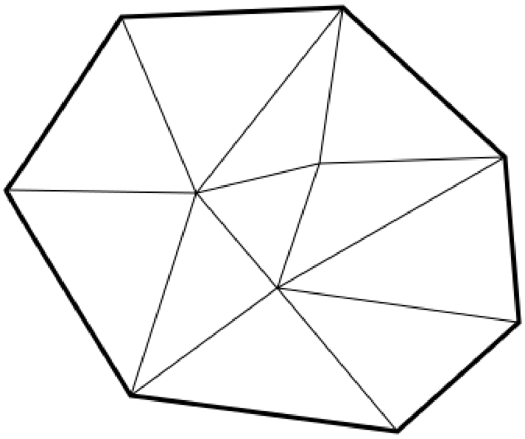

Figure 26: A **triangulation** of the domain.

We create a vector space of **piecewise linear** functions on the triangulated domain, each of which is linear over each of the triangles. The basis of this vector space $\{\phi_1, \phi_2, \ldots, \phi_N\}$ has cardinality N, and ϕ_i is defined as:

$$\phi_i = \begin{cases} 1 & \text{on } \boldsymbol{p}_i, \\ 0 & \text{on } \boldsymbol{p}_j \text{ for } i \neq j, \end{cases}$$

on the vertices, and is piecewise linear everywhere else, taking the value 0 on any triangle without $\boldsymbol{p}_i$ on its border, and taking a suitable linear value on any triangle with $\boldsymbol{p}_i$ on its border. See figure 27 for a diagram.

It should be clear that any piecewise linear function on Ω can be written as a linear combination of these ϕ_i.

A.3 Weak form

We write the problem in weak form, by first using Green's Theorem:

$$\int_\Omega \nabla^2 u \, v \, \mathrm{d}V + \int_\Omega \nabla u \cdot \nabla v \, \mathrm{d}V = \int_{\Gamma_D} (\nabla u \cdot \boldsymbol{n}) v \, \mathrm{d}s + \int_{\Gamma_N} \nabla u \cdot \boldsymbol{n} \, v \, \mathrm{d}s,$$

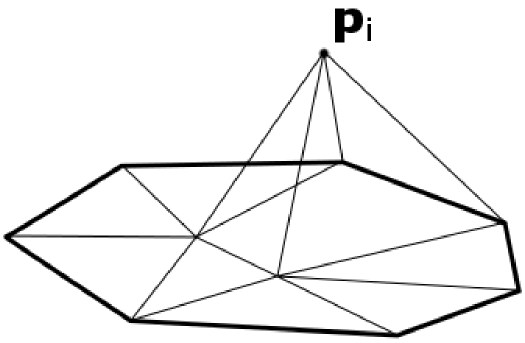

Figure 27: ϕ_i.

then substituting in the values we know from the formulation of the problem,

$$\int_\Omega \nabla u \cdot \nabla v \,\mathrm{d}V + c\int_\Omega uv \,\mathrm{d}V = \int_\Omega fv \,\mathrm{d}V + \int_{\Gamma_N} g_1 v \,\mathrm{d}s + \int_{\Gamma_D} \nabla u \cdot \boldsymbol{n}\, v \,\mathrm{d}s.$$

Take $v = 0$ on Γ_D, and we get the simple

$$\int_\Omega \nabla u \cdot \nabla v \,\mathrm{d}V + c\int_\Omega uv \,\mathrm{d}V = \int_\Omega fv \,\mathrm{d}V + \int_{\Gamma_N} g_1 v \,\mathrm{d}s \quad \forall v \text{ such that } v = 0 \text{ on } \Gamma_D.$$

A.4 Bringing it all together

Let V_D be the vector space of all piecewise linear functions on Ω that take the value 0 on Γ_D. The basis of this vector space is $\{\phi_i \,|\, \forall i \text{ such that } \boldsymbol{p}_i \text{ does not lie on } \Gamma_D\}$.

Recall the problem was this:

$$\begin{aligned} &\text{find } u, \\ &\text{such that } u = g_0 \text{ on } \Gamma_D, \text{ and} \\ \int_\Omega \nabla u \cdot \nabla v \,\mathrm{d}V + c\int_\Omega uv \,\mathrm{d}V = \int_\Omega fv \,\mathrm{d}V + \int_{\Gamma_N} g_1 v \,\mathrm{d}s \quad &\forall v \text{ such that } v = 0 \text{ on } \Gamma_D. \end{aligned}$$

We can discretise this into our vector space:

$$\begin{aligned} &\text{find } u \in V_D, \\ \text{such that } u(\boldsymbol{p}_i) = g_0(\boldsymbol{p}_i) \quad &\forall i \text{ such that } \boldsymbol{p}_i \in \Gamma_D, \text{ and} \\ \int_\Omega \nabla u \cdot \nabla v \,\mathrm{d}V + c\int_\Omega uv \,\mathrm{d}V = \int_\Omega fv \,\mathrm{d}V + \int_{\Gamma_N} g_1 v \,\mathrm{d}s \quad &\forall v \in V_D. \end{aligned}$$

We can actually simplify this to only the basis of V_D (it is again suggested that the reader read [4], on which this appendix is based, for a more thorough explanation of each step):

$$\begin{aligned}
&\text{find } u \in V_D,\\
&\text{such that } u(\boldsymbol{p}_i) = g_0(\boldsymbol{p}_i) \quad \forall i \text{ such that } \boldsymbol{p}_i \in \Gamma_D, \text{ and}\\
&\int_\Omega \nabla u \cdot \nabla \phi_i \,\mathrm{d}V + c\int_\Omega u\phi_i \,\mathrm{d}V = \int_\Omega f\phi_i \,\mathrm{d}V + \int_{\Gamma_N} g_1\phi_i \,\mathrm{d}s \quad \forall i \text{ such that } \boldsymbol{p}_i \notin \Gamma_D.
\end{aligned}$$

Finally, all of this can be written as a single linear system:

$$\begin{aligned}
&\sum_{j\in\mathrm{Ind}} \left(\int_\Omega \nabla\phi_j \cdot \nabla\phi_i \,\mathrm{d}V + c\int_\Omega \phi_j^2 \,\mathrm{d}V\right) u = \int_\Omega f\phi_i \,\mathrm{d}V + \int_{\Gamma_N} g_1\phi_i \,\mathrm{d}s\\
&-\sum_{j\in\mathrm{Dir}} \left(\int_\Omega \nabla\phi_j \cdot \nabla\phi_i \,\mathrm{d}V + c\int_\Omega \phi_j^2 \,\mathrm{d}V\right) g_0(\boldsymbol{p}_j),
\end{aligned}$$

where

$$\begin{aligned}
\mathrm{Ind} &= \{i \,|\, \forall i \text{ such that } \boldsymbol{p}_i \notin \Gamma_D\},\\
\mathrm{Dir} &= \{i \,|\, \forall i \text{ such that } \boldsymbol{p}_i \in \Gamma_D\}.
\end{aligned}$$

This is a linear system of as many equations as there are unknowns, and it is this that is solved in the finite element method.

This appendix was a grossly simplified version of [4], taking most of the steps from it, and skipping over many more. Refer to that work for more detail, rigour, and understanding.

B Code

All the code used in this project can be found in full on the author's GitHub page, `https://github.com/Candidate1008097/BSP_Code`. Some specifics will be covered here, using the code for the Schnackenberg problem (section 1.3) as an example.

Much of the code was supplied by this project's supervisor, however all of it has been edited to fit e.g. newer versions of the python library, FEniCS, different boundary conditions, and so on. The code for the combined models was written from scratch, with the provided code as reference.

B.1 FEniCS

FEniCS (`https://fenicsproject.org/`) is a python 2 library that uses the finite element method to solve partial differential equations in weak form. It prides itself on having the code representation of the weak form being very close to the actual mathematical expression, for example, compare

```
# weak form
Left  = u/dt * uT * fn.dx + v/dt * vT * fn.dx \
        + c1 * fn.inner(fn.grad(u), fn.grad(uT)) * fn.dx \
        + c2 * fn.inner(fn.grad(v), fn.grad(vT)) * fn.dx

Right = uold * uT/dt * fn.dx + vold * vT/dt * fn.dx \
        + d * (a - uold + uold * uold * vold) * uT * fn.dx \
        + d * (b - uold * uold * vold) * vT * fn.dx
```

to the weak form of the Schnackenberg problem,

$$\begin{aligned}
&\int_\Omega \frac{u^{n+1}}{\Delta t} u_{\text{test}}\,\mathrm{d}V + \int_\Omega \frac{v^{n+1}}{\Delta t} v_{\text{test}}\,\mathrm{d}V \\
&\qquad + c_1 \int_\Omega \nabla u^{n+1}\cdot\nabla u_{\text{test}}\,\mathrm{d}V + c_1 \int_\Omega \nabla v^{n+1}\cdot\nabla v_{\text{test}}\,\mathrm{d}V \\
&= \int_\Omega \frac{u^{n}}{\Delta t} u_{\text{test}}\,\mathrm{d}V + \int_\Omega \frac{v^{n}}{\Delta t} v_{\text{test}}\,\mathrm{d}V \\
&\qquad + d\int_\Omega \left(a - u^n + (u^n)^2 v^n\right) u_{\text{test}}\,\mathrm{d}V + d\int_\Omega \left(b - (u^n)^2 v^n\right) v_{\text{test}}\,\mathrm{d}V.
\end{aligned}$$

As you can imagine, this makes FEniCS a very powerful library.

B.2 Functions, function spaces, and constants

In FEniCS, one of the first things to do is define the function spaces we work over:

```
# function spaces
P1 = fn.FiniteElement("Lagrange", mesh.ufl_cell(), 1)
Mh = fn.FunctionSpace(mesh, "Lagrange", 1)
Nh = fn.FunctionSpace(mesh, fn.MixedElement([P1,P1]))
```

This utilises certain aspects of the finite element method we did not cover in our whistle-stop tour, so the reader is again encouraged to read [4]. Here we are defining the shape and degree of the interpolating functions.

We define our trial and test functions over these spaces like so:

```
# trial and test functions
u, v   = fn.TrialFunctions(Nh)
uT, vT = fn.TestFunctions(Nh)
```

and constants, to improve speed (as these computations can take a very long time!), are defined:

```
# model constants
a  = fn.Constant(0.1305)
b  = fn.Constant(0.7695)
c1 = fn.Constant(0.05)
c2 = fn.Constant(1.0)
d  = fn.Constant(170.0)
```

B.3 Time loops

For time-dependent problems, we create a time loop.

```
# time loop
inc = 0

while (t <= T):

    print ”t␣=”, t

    # solve
    BB = fn.assemble(Right)
    solver.solve(Sol.vector(), BB)

    # output to file
    u,v = Sol.split()
    if (inc % frequencySave == 0):
        u.rename(”u”,”u”); fileu << (u,t)
        v.rename(”v”,”v”); filev << (v,t)

    # update old values
    fn.assign(uold,u); fn.assign(vold,v)

    # increment
    t += dt; inc += 1
```

For a more complicated algorithm, like the one for our combined models at the end (recall the algorithm we defined), the time loop is equivalently more complicated:

```
# time loop
iteration = 0
```

```
while t <= TFinal:
    # log time
    print ”t␣=”, t

    # solve poroelasticity
    fn.solve(pe_left_hand_side == pe_right_hand_side,
             pe_solution,
             boundary_conditions)

    # split
    u, phi, p = pe_solution.split()

    # calculate sigma and z
    sigma = fn.project(2 * mu * strain(u_old) - phi * fn.Identity(2) +
            n_old * c_1 * fn.outer(f_0, f_0), Whf)
    z = fn.project((u - u_old)/dt, Vhf)

    rd_rhs_2 = rd_right_hand_side + Istim(t) * F * fn.dx

    # solve reaction-diffusion
    fn.solve(rd_left_hand_side == rd_rhs_2,
             rd_solution)

    # split
    E, n = rd_solution.split()

    # save if divisible by frequency
    if iteration % frequency == 0:
        u.rename(”u”, ”u”)
        phi.rename(”phi”, ”phi”)
        p.rename(”p”, ”p”)
        E.rename(”E”, ”E”)
        n.rename(”n”, ”n”)
        pvdU << (u, t)
        pvdPHI << (phi, t)
        pvdP << (p, t)
        pvdE << (E, t)
        pvdN << (n, t)

    # update solutions
    fn.assign(u_old, u)
    fn.assign(phi_old, phi)
    fn.assign(p_old, p)
    fn.assign(E_old, E)
    fn.assign(n_old, n)

    # update time
    t += dt
    iteration += 1
```

The reader is encouraged to compare this time loop to the algorithm:

1. Define $\boldsymbol{u}^0$, ϕ^0, p^0, E^0, and n^0, the initial values of our variables. (We will set them all to 0 everywhere.) Set $t = 0$ and $n = 0$.

2. While $t < T$, our final time:

   (a) Use the poroelasticity equations, using the values ϕ^n, p^n, and n^n, to solve the system for $\boldsymbol{u}^{n+1}$, ϕ^{n+1}, and p^{n+1}. (We do not need $\boldsymbol{u}^n$, as it does not appear in the required equations.)

   (b) Find the new value of $\boldsymbol{\sigma}^n_{\text{total}}$ from ϕ^{n+1}, p^{n+1}, and n^n (as we don't know n^{n+1} yet), and calculate an approximation for the medium velocity $\boldsymbol{z}^n$ as $(\boldsymbol{u}^{n+1} - \boldsymbol{u}^n)/\Delta t$.

   (c) Solve the reaction-diffusion problem with the Karma equations, finding E^{n+1} and n^{n+1} from E^n, n^n, $\boldsymbol{z}^n$, and $\boldsymbol{\sigma}^n_{\text{total}}$.

   (d) Increase n by 1, and t by Δt.

References

[1] L. Berger, R. Bordas, D. Kay, and S. Tavener (2017).
A stabilized finite element method for finite-strain three-field poroelasticity.
Computational Mechanics, 60(1), pp. 51–68.

[2] A. Karma (1994).
Electrical alternans and spiral wave breakup in cardiac tissue.
Chaos: An Interdisciplinary Journal of Nonlinear Science, 4(3), pp. 461–472.

[3] R. Oyarzúa and R. Ruiz-Baier (2016).
Locking-free finite element methods for poroelasticity.
SIAM Journal on Numerical Analysis, 54(5), pp. 2951–2973.

[4] F.-J. Sayas (2008).
A gentle introduction to the finite element method.
Lecture notes, University of Delaware.
http://www.math.udel.edu/~fjsayas/documents/anIntro2FEM_2015.pdf

[5] S. Terragni (2013).
Poroelastic computational modeling of biological tissues. Application to the mechanics of the eye.
http://hdl.handle.net/10589/88368

[6] S. Whitaker (1986).
Flow in porous media I: A theoretical derivation of Darcy's law.
Transport in Porous Media, 1(1), pp. 3–25.

[7] A.-T. Vuong, L. Yoshihara, and W. A. Wall (2014).
A general approach for modeling interacting flow through porous media under finite deformations.
Computer Methods in Applied Mechanics and Engineering, 283, pp. 1240–1259.