Spatial Modelling of Lesion Development Informed by Multiple Sclerosis Patient MRIs

Geneva Birtles

Supervised by Dr Adrianne Jenner, Dr Robyn Araujo and Dr Michael Dallaston
Queensland University of Technology

Contents

Abstract

This paper investigates the mathematical modelling of lesions in Multiple Sclerosis patients, spatially and over time. No previous research had been identified that attempted to model this particular immune response on a cellular level using mathematics. This paper outlines the key components and dynamics that are captured by the model, and the formulation of a bespoke system of Partial Differential Equations which were then solved using the Finite-Volume method. MATLAB programming was utilised to simulate the system over time, and the behaviour of the model could then be visualised. Our findings indicate that, based on the preliminary PDE model created, all populations of myelin converged to a steady-state of full regeneration over time. Suggestions for further research into this model have been made, namely parameterisation and improving the model to capture the realistic nature of T-cell activity observed in Multiple Sclerosis patients.

1 Introduction

Multiple Sclerosis (MS) is an autoimmune, demyelinating and neurodegenerative disease that affects the brain and central nervous system [1]. All nerve cells have a protective, insulating cover called a myelin sheath [1]. This particular demyelinating disease describes the case in which myelin present in the brain and spinal cord gets degraded and exhibits damage. In MS patients, damage to this myelin occurs when the immune system mistakenly attacks the myelin causing scarring in the brain [2]; these scars are referred to as lesions and are visible on MRI scans. As lesions form within the brain and/or spinal cord, they disrupt the ability of the affected nerve cells to transmit signals which can result in a range of neurological symptoms, depending on the location of these lesions [1]. Little is known about the possible causes or triggers of MS and currently there is no cure for the disease, only treatments to reduce symptoms.

The key objective of this paper is to model the immune response on a cellular level that is triggered by demyelination in the brain using a system of Partial Differential Equations (PDEs). More broadly, our aim is to create a model that can predict the evolution of a patient's disease throughout their lifetime and from this, identify optimal intervention points for treatment.

We identified four main components to simulate with this model that are imperative to the immune response: healthy myelin (M), damaged myelin (M_D), chemokine (C) and T-cells (T). Damaged myelin initially present in the domain triggers an immune response. Firstly, the lesions will secrete chemokine, which is also known as a chemotactic cytokine [2]. This chemokine is the main player in the immune response as it is a protein that acts as an attractant to immune cells. This leads us to the action of chemotaxis, which is the movement of immune cells (namely T-cells) towards the chemoattractant (chemokine) as the signalling protein induces cell migration to area of damage [3]. Thus, we will also be modelling T-cells, a type of CD8+ immune cell, that will travel up the chemokine gradient towards lesions and attempt to heal them [4].

This paper outlines the formulation of the model as a system of four PDEs and the mathematical discretisation of these PDEs (spatially and temporally) according to the Finite-Volume Method. This paper also explains the implementation of solving and visualising this system using the programming language MATLAB and outlines the key findings and areas for further research.

2   Statement of Authorship

The contributors to this academic paper include the author Geneva Birtles and supervisors Adrianne Jenner, Robyn Araujo and Michael Dallaston. Adrianne Jenner originally conceived the project outline and objectives, and provided specialist knowledge of Multiple Sclerosis. Geneva Birtles and Adrianne Jenner formulated the system of partial differential equations. Geneva Birtles performed the Finite-Volume Method discretisation and MATLAB computational work with assistance from all supervisors. Geneva Birtles ran all computational simulations and analysed results with support from supervisors.

3   Modelling the Immune Response

3.1   Modelling Chemotaxis

Initially, we can formulate the PDE models that describe the chemotaxis of T-cells up the chemokine gradient. This model has been based off the the minimal model derived in article [3], defined in Equations 1 and 2 below:

$$\frac{\partial v}{\partial t} = \nabla^2 v + u - v \tag{1}$$

$$\frac{\partial u}{\partial t} = \nabla(D\nabla u - \chi u \nabla v) \tag{2}$$

Where u is chemotaxing towards the signal v, t is time and ∇ represents the two-dimensional gradient operator.

Thus, we will have the chemokine acting as the signal v, exhibiting simple diffusion (as seen in Equation (1)) and will have the T-cells (denoted u in the minimal model) diffusing and chemotaxing towards this signal (as seen in Equation (2)). Evidently, we can derive the two-dimensional PDEs for chemokine and T-cells, by expanding the gradient terms and defining the relevant constants.

Preliminary Chemokine Model (2D Diffusion):

$$\frac{\partial C(x,t)}{\partial t} = \nabla^2(D_c\, C(x,t))$$

$$\frac{\partial C(x,t)}{\partial t} = \nabla \cdot (D_c\, \nabla C(x,t)) \tag{3}$$

$$\frac{\partial C(x,t)}{\partial t} = \nabla \cdot \left(D_c\, \frac{\partial C(x,t)}{\partial x}\mathbf{i} + D_c\, \frac{\partial C(x,t)}{\partial y}\mathbf{j} \right)$$

$$\frac{\partial C(x,t)}{\partial t} = \frac{\partial}{\partial x}\left(D_c\, \frac{\partial C(x,t)}{\partial x} \right) + \frac{\partial}{\partial y}\left(D_c\, \frac{\partial C(x,t)}{\partial y} \right) \tag{4}$$

Where the chemokine diffusion constant is defined by D_c and we do not yet have any additional source or sink terms.

Preliminary T-cell Model (2D Diffusion and Chemotaxis):

$$\frac{\partial T(x,t)}{\partial t} = \nabla \cdot (D_t \nabla T(x,t) - \chi T(x,t) \nabla C(x,t)) \tag{5}$$

$$\frac{\partial T(x,t)}{\partial t} = \nabla \cdot \left(\left(D_t \frac{\partial T(x,t)}{\partial x}\mathbf{i} + D_t \frac{\partial T(x,t)}{\partial y}\mathbf{j} \right) - \left(\chi T(x,t) \frac{\partial C(x,t)}{\partial x}\mathbf{i} + \chi T(x,t) \frac{\partial C(x,t)}{\partial y}\mathbf{j} \right) \right)$$

$$\frac{\partial T(x,t)}{\partial t} = \nabla \cdot \left(D_t \frac{\partial T(x,t)}{\partial x}\mathbf{i} + D_t \frac{\partial T(x,t)}{\partial y}\mathbf{j} \right) - \nabla \cdot \left(\chi T(x,t) \frac{\partial C(x,t)}{\partial x}\mathbf{i} + \chi T(x,t) \frac{\partial C(x,t)}{\partial y}\mathbf{j} \right)$$

$$\frac{\partial T(x,t)}{\partial t} = \frac{\partial}{\partial x}\left(D_t \frac{\partial T(x,t)}{\partial x} \right) + \frac{\partial}{\partial y}\left(D_t \frac{\partial T(x,t)}{\partial y} \right) - \frac{\partial}{\partial x}\left(\chi T(x,t) \frac{\partial C(x,t)}{\partial x} \right) - \frac{\partial}{\partial y}\left(\chi T(x,t) \frac{\partial C(x,t)}{\partial y} \right)$$

$$\frac{\partial T(x,t)}{\partial t} = \frac{\partial}{\partial x}\left(D_t \frac{\partial T(x,t)}{\partial x} - \chi T(x,t) \frac{\partial C(x,t)}{\partial x} \right) + \frac{\partial}{\partial y}\left(D_t \frac{\partial T(x,t)}{\partial y} - \chi T(x,t) \frac{\partial C(x,t)}{\partial y} \right) \tag{6}$$

Where the T-cell diffusion constant is defined by D_t, the chemotactic sensitivity (rate of chemotaxis) is defined by χ and we do not yet have any additional source or sink terms.

3.2 Formulation of the System of Coupled Partial Differential Equations

Continuing from the preliminary chemotaxis PDEs, we can begin to formulate our full system of equations for each of the four components. These PDEs model the concentrations of each of these substances in the domain over time. Both concentrations of healthy myelin (M) and damaged myelin (M_D) are nondimensionalised by nature and can vary from 0 to 1. See Figure 1 for full visualisation of system dynamics and interactions between each of the components.

Healthy Myelin:

The healthy myelin grows logistically in the domain (at rate, r) and becomes damaged by nearby damaged myelin (at rate v). Healthy myelin begins at a homogenous profile over the entire domain and the carrying capacity of healthy myelin plus damaged myelin is 1. The PDE to model healthy myelin is defined below in Equation 7.

$$\frac{\partial M(x,t)}{\partial t} = rM(1 - (M + M_D)) - vMM_D \tag{7}$$

Damaged Myelin:

The damaged myelin diffuses throughout the domain (at rate D_M), damaging neighboring healthy myelin (at rate v) and growing in the presence of T-cells (at rate δ). This damaged myelin represents lesions within the domain. The PDE to model damaged myelin is defined below in Equation 8.

$$\frac{\partial M_D(x,t)}{\partial t} = D_M \nabla^2 M_D + vMM_D - \delta M_D T \tag{8}$$

Chemokine:

The chemokine diffuses throughout the domain (at rate D_C) and is secreted by the damaged myelin (at rate s); the chemokine also decays at rate δ_C. The PDE to model chemokine activity is defined below in Equation 9.

$$\frac{\partial C(x,t)}{\partial t} = D_C \nabla^2 C - \delta_C C + s M_D \tag{9}$$

T-cells:

T-cells diffuse throughout the domain (at rate D_t) while simultaneously chemotaxing up the chemokine gradient (at rate χ); the T-cells also decay at rate δ_T. The PDE to model T-cell activity is defined below in Equation 10.

$$\frac{\partial T(x,t)}{\partial t} = \nabla \cdot (D_t \nabla T - \chi T \nabla C) - \delta_T T \tag{10}$$
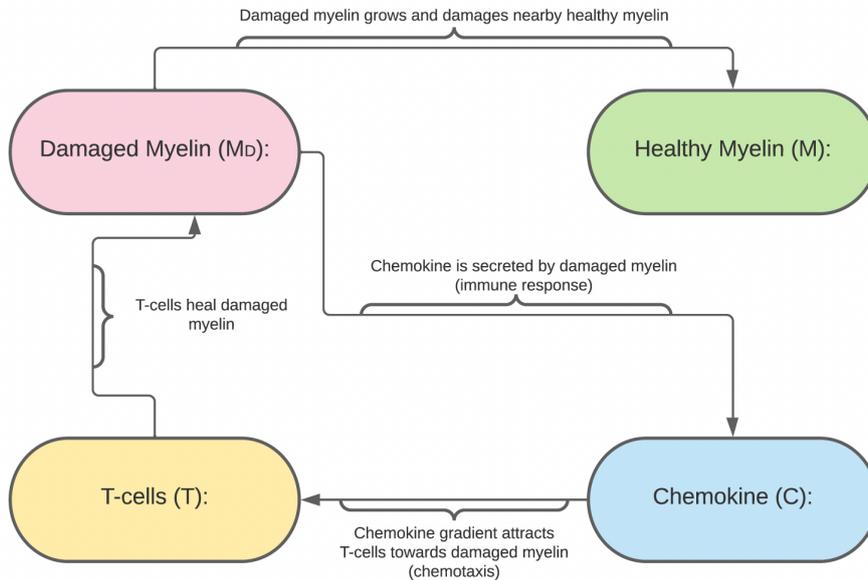
Figure 1: System Dynamics and Interactions

4   Numerical Solving Method

Due to the complexity of the system of PDEs, it is virtually impossible to solve them using analytical methods. Thus, the Finite Volume Method was chosen to solve the system numerically for each of the four components at predefined discrete points. This method will involve discretising the PDEs spatially and temporally to formulate a simpler system of equations and then inputting these into MATLAB to be solved computationally at discrete time points. This method will be detailed in the following sections.

5

4.1 Finite-Volume Method

The Finite-Volume method (FVM) works by manipulating the existing PDEs into a system of simpler ODEs that can be solved at a set of discrete time points. Firstly, the spatial domain must be predefined and split up into a mesh of discrete points (termed nodes), each of which lies at the centre of a rectangular control volume, which in total make up the entire domain. A spatial discretisation can then be performed for each of the PDEs at these nodes. This works by integrating the PDEs over the control volume and converting the divergence terms into surface integrals, using the divergence theorem. We can then evaluate these surface integrals as fluxes at each of the four sides of the control volume. The Finite-Volume method is conservative as the flux entering a particular control volume is equal to the flux leaving the surrounding control volumes. After the spatial discretisation has been executed, a temporal discretisation can be conducted to in order to simulate the solution over time. The *Theta Method* is utilised in this time discretisation, which works by integrating the resultant ODE (from the spatial discretisation) from $t = t_n$ to $t = t_{n+1}$ and approximating the resulting integrals using the weighted theta (θ) approximation. From this, each of these equations can be solved computationally using a Newton solver in MATLAB.

4.2 Spatial Discretisation

We can now implement the spatial discretisation of our defined system of PDEs (Equations 8, 9 and 10), since the PDE for healthy myelin doesn't contain a flux term (Equation 7) it doesn't require a spatial discretisation. Initially, we will be implementing a simple no-flux boundary condition for our PDEs:

$$\frac{\partial C(x,t)}{\partial n} = 0, \quad \text{on} \quad \partial\Omega, t > 0 \tag{11}$$

$$\frac{\partial M_D(x,t)}{\partial n} = 0, \quad \text{on} \quad \partial\Omega, t > 0 \tag{12}$$

$$\frac{\partial T(x,t)}{\partial n} = 0, \quad \text{on} \quad \partial\Omega, t > 0 \tag{13}$$

Where $\frac{\partial C(x,t)}{\partial n}$, $\frac{\partial M_D(x,t)}{\partial n}$ and $\frac{\partial T(x,t)}{\partial n}$ denote the directional derivative normals to the boundary $\partial\Omega$.

4.2.1 Chemokine PDE

To begin our spatial discretisation of the chemokine PDE (9), we can rearrange equation (3) and integrate over the control volume (denoted V_p), such that:

$$\frac{\partial C(x,t)}{\partial t} = \nabla \cdot (D_c \nabla C(x,t)) - \delta_C C + s M_D$$

$$0 = \frac{\partial C(x,t)}{\partial t} + \nabla \cdot (-D_c \nabla C(x,t)) + \delta_C C - s M_D$$

$$0 = \int\int_{V_p} \frac{\partial C(x,t)}{\partial t} dV + \int\int_{V_p} \nabla \cdot \mathbf{J} \, dV + \delta_C \int\int_{V_p} C(x,t) \, dV - s \int\int_{V_p} M_D(x,t) \, dV \tag{14}$$

Where:

$$\mathbf{J} = -D_c \nabla C(x,t) \tag{15}$$

We can then introduce a control volume averaged value of C, $\frac{\partial C}{\partial t}$ and M_D over V_p (see Figure 2 for visualisation of node design):

$$\bar{C}_P = \frac{1}{\Delta x_P \Delta y_P} \int\int_{V_p} C(x,t) dV$$

$$\frac{d\bar{C}_P(t)}{dt} = \frac{1}{\Delta x_P \Delta y_P} \int\int_{V_p} \frac{\partial C(x,t)}{\partial t} dV$$

$$\bar{M}_{DP} = \frac{1}{\Delta x_P \Delta y_P} \int\int_{V_p} M_D(x,t) dV$$

These can subsequently be rearranged:

$$\int\int_{V_p} C(x,t) dV = \Delta x_P \Delta y_P \, \bar{C}_P$$

$$\int\int_{V_p} \frac{\partial C(x,t)}{\partial t} dV = \Delta x_P \Delta y_P \, \frac{d\bar{C}_P(t)}{dt}$$

$$\int\int_{V_p} M_D(x,t) dV = \Delta x_P \Delta y_P \, \bar{M}_{DP}$$

These approximations can then be substituted into Equation 14:

$$0 = \Delta x_P \Delta y_P \, \frac{d\bar{C}_P(t)}{dt} + \int\int_{V_p} \nabla \cdot \mathbf{J} dV + \delta_C \Delta x_P \Delta y_P \, \bar{C}_P - s \Delta x_P \Delta y_P \, \bar{M}_{DP}$$

Dividing through by $\Delta x_P \Delta y_P$ gives:

$$0 = \frac{d\bar{C}_P(t)}{dt} + \frac{1}{\Delta x_P \Delta y_P} \int\int_{V_p} \nabla \cdot \mathbf{J} dV + \delta_C \bar{C}_P - s\bar{M}_{DP} \tag{16}$$

We can then approximate the remaining integral using Green's Theorem. This allows the double integral over the two-dimensional divergence ($\nabla \cdot \mathbf{J}$) over the control volume to be expressed as follows, Where, $\hat{\mathbf{n}}$ is the outward unit normal on the boundary of the control volume (Γ_p). As our control volume is rectangular in shape, this can be further separated into the four distinct faces (east, north, west, south), such that $\mathcal{E}_p = \{e, n, w, s\}$. Thus, we can state:

$$\int\int_{V_p} \nabla \cdot \mathbf{J} dV = \oint_{\Gamma_p} \mathbf{J} \cdot \hat{\mathbf{n}} \, d\sigma = \sum_{j \in \mathcal{E}_p} \int_{\Gamma_j} \mathbf{J} \cdot \hat{\mathbf{n}} \, d\sigma$$

As this line integral cannot be computed by analytical techniques we can employ a numerical strategy: a one-point midpoint quadrature approximation. Such that, we can define $\mathbf{m}_j$ as the coordinated of the midpoint of the cell face Γ_j and correspondingly, defining l_j as the cell face length. Thus, we can use the following approximation:

$$\int_{\Gamma_j} \mathbf{J} \cdot \hat{\mathbf{n}} \, d\sigma \approx (\mathbf{J} \cdot \hat{\mathbf{n}})_{\mathbf{m}_j} l_j$$

Which gives:

$$\int\int_{V_p} \nabla \cdot \mathbf{J} dV \approx \sum_{j \in \mathcal{E}_p} (\mathbf{J} \cdot \hat{\mathbf{n}})_{\mathbf{m}_j} l_j$$

For each of our four cell faces we can define the corresponding unit normals and face lengths, as follows:

East Face ($j = e$): $\hat{\mathbf{n}} = \mathbf{i}$ and $l_e = \Delta y_P$, which gives $(\mathbf{J} \cdot \hat{\mathbf{n}})_{\mathbf{m}_e} l_e = J_e \Delta y_P$

West Face ($j = w$): $\hat{\mathbf{n}} = \textbf{-i}$ and $l_w = \Delta y_P$, which gives $(\mathbf{J} \cdot \hat{\mathbf{n}})_{\mathbf{m}_w} l_w = -J_w \Delta y_P$

North Face ($j = n$): $\hat{\mathbf{n}} = \mathbf{j}$ and $l_n = \Delta x_P$, which gives $(\mathbf{J} \cdot \hat{\mathbf{n}})_{\mathbf{m}_n} l_n = J_n \Delta x_P$

South Face ($j = s$): $\hat{\mathbf{n}} = \textbf{-j}$ and $l_s = \Delta x_P$, which gives $(\mathbf{J} \cdot \hat{\mathbf{n}})_{\mathbf{m}_s} l_s = -J_s \Delta x_P$

Thus, our double integral can be rewritten finally as:

$$\int \int_{V_p} \nabla \cdot \mathbf{J} dV \approx \Delta y_P (J_e - J_w) + \Delta x_P (J_n - J_s)$$

This approximation can then be substituted back into Equation 16:

$$0 = \frac{d\bar{C}_P(t)}{dt} + \frac{1}{\Delta x_P \Delta y_P}(\Delta y_P(J_e - J_w) + \Delta x_P(J_n - J_s)) + \delta_C \bar{C}_P - s\bar{M}_{DP}$$

$$0 = \frac{d\bar{C}_P(t)}{dt} + \frac{1}{\Delta x_P}(J_e - J_w) + \frac{1}{\Delta y_P}(J_n - J_s) + \delta_C \bar{C}_P - s\bar{M}_{DP} \tag{17}$$

Using Equation 15 and a standard forward difference approximation for the partial derivatives, we can now evaluate our J terms:

$$J_e = -D_c \left[\frac{\partial C}{\partial x} \right]_e \approx -D_c \left(\frac{C_E - C_P}{\delta x_e} \right)$$

$$J_w = -D_c \left[\frac{\partial C}{\partial x} \right]_w \approx -D_c \left(\frac{C_P - C_W}{\delta x_w} \right)$$

$$J_n = -D_c \left[\frac{\partial C}{\partial y} \right]_n \approx -D_c \left(\frac{C_N - C_P}{\delta y_n} \right)$$

$$J_s = -D_c \left[\frac{\partial C}{\partial y} \right]_s \approx -D_c \left(\frac{C_P - C_S}{\delta y_s} \right)$$

Where C_P is the current concentration of chemokine at our particular node (node we are currently evaluating), C_E is the concentration at the node to the right of our particular node, C_W is the concentration at the node to the left, C_N is the concentration at the node above and C_S is the concentration at the node below (see Figure 2 for full visualisation of node design).

Hence, substituting our J values into Equation 17 and letting $\bar{C}_P \approx C_P$ and $\bar{M}_{DP} \approx M_{DP}$, we can derive our full Finite Volume Equation (FVE):

$$0 = \frac{dC_P(t)}{dt} + \frac{D_c}{\Delta x_P} \left(\frac{C_P - C_W}{\delta x_w} - \frac{C_E - C_P}{\delta x_e} \right) + \frac{D_c}{\Delta y_P} \left(\frac{C_P - C_S}{\delta y_s} - \frac{C_N - C_P}{\delta y_n} \right) + \delta_C C_P - s M_{DP} \tag{18}$$

Boundary Conditions:

For this chemokine PDE we assume no-flux boundary conditions along all sides of the domain (see Equation 11) we can derive alternative Finite Volume Equations (adapted from Equation 17) accordingly for each boundary node. See Appendix A for full set of boundary Finite Volume Equations.
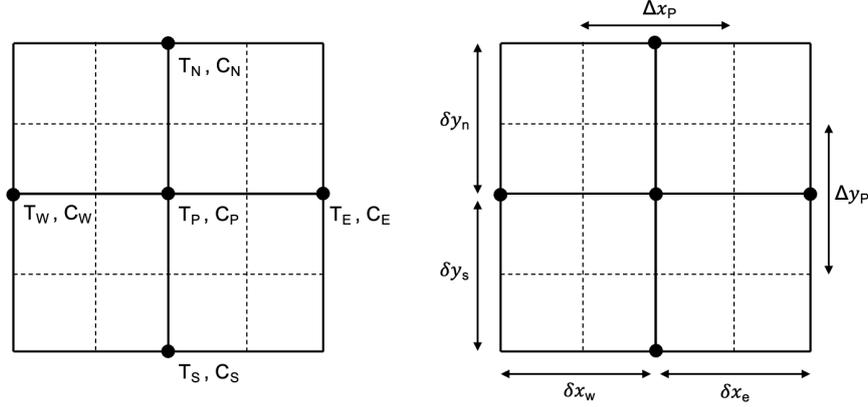
Figure 2: Node/Mesh Design

4.2.2   T-cell PDE

Following the same derivation method of the FVE used in section 4.2.1, we take Equation 5 and integrate over the control volume, such that:

$$\frac{\partial T(x,t)}{\partial t} = \nabla \cdot (D_t \nabla T(x,t) - \chi T(x,t) \nabla C(x,t)) - \delta_T T$$

$$0 = \int \int_{V_p} \frac{\partial T(x,t)}{\partial t} dV + \int \int_{V_p} \nabla \cdot \mathbf{Q} \, dV + \int \int_{V_p} \delta_T T \, dV \tag{19}$$

where

$$\mathbf{Q} = \chi T(x,t) \nabla C(x,t) - D_t \nabla T(x,t) \tag{20}$$

Using the relevant control volume averaged values of T and $\frac{\partial T}{\partial t}$ over V_p, this can be simplified:

$$0 = \frac{d\bar{T}_P(t)}{dt} + \frac{1}{\Delta x_P \Delta y_P} \int \int_{V_p} \nabla \cdot \mathbf{Q} \, dV + \delta_T \bar{T}_P$$

Again using Green's Theorem to evaluate the second double integral, we arrive at the following approximation:

$$\int \int_{V_p} \nabla \cdot \mathbf{Q} \, dV \approx \Delta y_P (Q_e - Q_w) + \Delta x_P (Q_n - Q_s)$$

Thus,

$$0 = \frac{d\bar{T}_P(t)}{dt} + \frac{1}{\Delta x_P}(Q_e - Q_w) + \frac{1}{\Delta y_P}(Q_n - Q_s) + \delta_T \bar{T}_P \tag{21}$$

Using Equation 20, a standard forward difference approximation for the partial derivatives and an average approximation of our T terms (see Table 1 for full list of approximations), we can now evaluate our Q terms. It is also useful to utilise the rewritten form of our PDE (see Equation 6).

$$Q_e = \chi [T]_e \left[\frac{\partial C}{\partial x} \right]_e - D_t \left[\frac{\partial T}{\partial x} \right]_e \approx \chi \left(\frac{T_E + T_P}{2} \right) \left(\frac{C_E - C_P}{\delta x_e} \right) - D_t \left(\frac{T_E - T_P}{\delta x_e} \right)$$

$$Q_w = \chi [T]_w \left[\frac{\partial C}{\partial x} \right]_w - D_t \left[\frac{\partial T}{\partial x} \right]_w \approx \chi \left(\frac{T_W + T_P}{2} \right) \left(\frac{C_P - C_W}{\delta x_w} \right) - D_t \left(\frac{T_P - T_W}{\delta x_w} \right)$$

9

$$Q_n = \chi [T]_n \left[\frac{\partial C}{\partial y}\right]_n - D_t \left[\frac{\partial T}{\partial y}\right]_n \approx \chi \left(\frac{T_N + T_P}{2}\right)\left(\frac{C_N - C_P}{\delta y_n}\right) - D_t \left(\frac{T_N - T_P}{\delta y_n}\right)$$

$$Q_s = \chi [T]_s \left[\frac{\partial C}{\partial y}\right]_s - D_t \left[\frac{\partial T}{\partial y}\right]_s \approx \chi \left(\frac{T_S + T_P}{2}\right)\left(\frac{C_P - C_S}{\delta y_s}\right) - D_t \left(\frac{T_P - T_S}{\delta y_s}\right)$$

Where T_P is the current concentration of T-cells at our particular node, T_E is the concentration at the node to the right of our particular node, T_W is the concentration at the node to the left, T_N is the concentration at the node above and T_S is the concentration at the node below (see Figure 2 for full visualisation of nodal design).

Hence, substituting our Q values into Equation 21 and letting $T_P \approx \bar{T}_P$ we can derive our full Finite Volume Equation:

$$
\begin{aligned}
0 = \frac{dT_P(t)}{dt} + \frac{1}{\Delta x_P}\Bigg[& \chi\left(\frac{T_E + T_P}{2}\right)\left(\frac{C_E - C_P}{\delta x_e}\right) - D_t\left(\frac{T_E - T_P}{\delta x_e}\right) - \chi\left(\frac{T_W + T_P}{2}\right)\left(\frac{C_P - C_W}{\delta x_w}\right) + \\
& D_t\left(\frac{T_P - T_W}{\delta x_w}\right)\Bigg] + \frac{1}{\Delta y_P}\Bigg[\chi\left(\frac{T_N + T_P}{2}\right)\left(\frac{C_N - C_P}{\delta y_n}\right) - D_t\left(\frac{T_N - T_P}{\delta y_n}\right) - \\
& \chi\left(\frac{T_S + T_P}{2}\right)\left(\frac{C_P - C_S}{\delta y_s}\right) + D_t\left(\frac{T_P - T_S}{\delta y_s}\right)\Bigg] + \delta_T T_P
\end{aligned}
\tag{22}
$$

Table 1: Relevant Approximations (Central Difference & Averaging)

Relevant Approximations		
$\left[\frac{\partial T}{\partial y}\right]_n \approx \frac{T_N - T_P}{\delta y_n}$	$\left[\frac{\partial C}{\partial y}\right]_n \approx \frac{C_N - C_P}{\delta y_n}$	$[T]_n \approx \frac{T_N + T_P}{2}$
$\left[\frac{\partial T}{\partial y}\right]_s \approx \frac{T_P - T_S}{\delta y_s}$	$\left[\frac{\partial C}{\partial y}\right]_s \approx \frac{C_P - C_S}{\delta y_s}$	$[T]_s \approx \frac{T_S + T_P}{2}$
$\left[\frac{\partial T}{\partial x}\right]_e \approx \frac{T_E - T_P}{\delta x_e}$	$\left[\frac{\partial C}{\partial x}\right]_e \approx \frac{C_E - C_P}{\delta x_e}$	$[T]_e \approx \frac{T_E + T_P}{2}$
$\left[\frac{\partial T}{\partial x}\right]_w \approx \frac{T_P - T_W}{\delta x_w}$	$\left[\frac{\partial C}{\partial x}\right]_w \approx \frac{C_P - C_W}{\delta x_w}$	$[T]_w \approx \frac{T_W + T_P}{2}$

Boundary Conditions:

For this T-cell PDE we assume no-flux boundary conditions along all sides of the domain (see Equation 13) we can derive alternative Finite Volume Equations (adapted from Equation 21) accordingly for each boundary node. See Appendix B for full set of boundary Finite Volume Equations.

4.2.3 Damaged Myelin PDE

Given the similarity of the damaged myelin PDE to the chemokine PDE the spatial discretisation follows the same process; the full derivation of the Finite Volume Equations can be found in Appendix C. The resulting general Finite Volume Equation is as follows:

$$
\begin{aligned}
0 = \frac{dM_{DP}(t)}{dt} + \frac{D_m}{\Delta x_P}\left(\frac{M_{DP} - M_{DW}}{\delta x_w} - \frac{M_{DE} - M_{DP}}{\delta x_e}\right) + \frac{D_m}{\Delta y_P}\Bigg(\frac{M_{DP} - M_{DS}}{\delta y_s} \\
- \frac{M_{DN} - M_{DP}}{\delta y_n}\Bigg) - vM_P M_{DP} - \delta M_{DP} T_P
\end{aligned}
\tag{23}
$$

10

4.3 Temporal Discretisation

Thus, as we have completed all spatial discretisation for the system, we can now discretise our system in time utilising the Theta method. In order to do this we can first integrate our FVEs (Equations 21, 21 and 23) and our PDE for healthy myelin (Equation 7) from $t = t_n$ to $t = t_{n+1}$ and approximate the resulting integrals using the weighted θ approximation, defined as follows:

$$\int_{t_n}^{t_{n+1}} f(t)dt \approx \delta t[(1-\theta)f(t_n) + \theta f(t_{n+1})] \tag{24}$$

Starting with the T-cell FVE, we can integrate the Finite Volume Equation 21 from $t = t_n$ to $t = t_{n+1}$, which gives:

$$0 = \int_{t_n}^{t_{n+1}} \frac{dT_P(t)}{dt}dt + \int_{t_n}^{t_{n+1}} \frac{1}{\Delta x_P}(Q_e - Q_w)dt + \int_{t_n}^{t_{n+1}} \frac{1}{\Delta y_P}(Q_n - Q_s)dt + \int_{t_n}^{t_{n+1}} \delta_T T_P dt$$

Evaluating the time derivative term:

$$\int_{t_n}^{t_{n+1}} \frac{dT_P(t)}{dt}dt = [T_P(t)]_{t_n}^{t_{n+1}} = T_P^{(n+1)} - T_P^{(n)}$$

Approximating remaining terms using the weighted theta approximation (24) gives:

$$\int_{t_n}^{t_{n+1}} \frac{1}{\Delta x_P}(Q_e - Q_w)dt \approx \frac{\delta t}{\Delta x_P}[(1-\theta_T)(Q_e^{(n)} - Q_w^{(n)}) + \theta_T(Q_e^{(n+1)} - Q_w^{(n+1)})]$$

$$\int_{t_n}^{t_{n+1}} \frac{1}{\Delta y_P}(Q_n - Q_s)dt \approx \frac{\delta t}{\Delta y_P}[(1-\theta_T)(Q_n^{(n)} - Q_s^{(n)}) + \theta_T(Q_n^{(n+1)} - Q_s^{(n+1)})]$$

$$\int_{t_n}^{t_{n+1}} \delta_T T_P dt \approx \delta t \delta_T[(1-\theta_T)T_P^{(n)} + \theta_T T_P^{(n+1)}]$$

Finally, implementing the *Backward Euler* Method by taking $\theta_T = 1$ and substituting these back into our FVEs we obtain:

T-cell Discretisation:

$$0 = T_P^{(n+1)} - T_P^{(n)} + \delta t \left[\frac{1}{\Delta x_P}\left(Q_e^{(n+1)} - Q_w^{(n+1)}\right) + \frac{1}{\Delta y_P}\left(Q_n^{(n+1)} - Q_s^{(n+1)}\right) + \delta_T T_P^{(n+1)} \right] \tag{25}$$

Repeating this method for each of the other PDEs, gives the follow results:

Chemokine Discretisation:

$$0 = C_P^{(n+1)} - C_P^{(n)} + \delta t \left[\frac{1}{\Delta x_P}\left(J_e^{(n+1)} - J_w^{(n+1)}\right) + \frac{1}{\Delta y_P}\left(J_n^{(n+1)} - J_s^{(n+1)}\right) + \delta_C C_P^{(n+1)} - s M_{D_P}^{(n+1)} \right] \tag{26}$$

Damaged Myelin Discretisation:

$$0 = M_{D_P}^{(n+1)} - M_{D_P}^{(n)} + \delta t \left[\frac{1}{\Delta x_P}\left(W_e^{(n+1)} - W_w^{(n+1)}\right) + \frac{1}{\Delta y_P}\left(W_n^{(n+1)} - W_s^{(n+1)}\right) \right.$$
$$\left. - v M_P^{(n+1)} M_{D_P}^{(n+1)} - \delta M_{D_P}^{(n+1)} T_P^{(n+1)} \right] \tag{27}$$

Healthy Myelin Discretisation:

$$0 = M_P^{(n+1)} - M_P^{(n)} - \delta t \left[r M_P^{(n+1)}(1 - (M_P^{(n+1)} + M_{D_P}^{(n+1)})) - v M_P^{(n+1)} M_{D_P}^{(n+1)} \right] \tag{28}$$

11

5   Computational Implementation

The fully discretised FVM equations (Equations 25 - 28) could then be solved on MATLAB by iterating through discrete timesteps. Initially, a newton solver code was written to solve the system but was later replaced with an inbuilt MATLAB function *'fsolve'* in order to improve efficiency. As the system was not completely nondimensionalised and no parameter estimation had been conducted, arbitrary stand-in parameter values were used for simulations. Running simulations for 1500 timesteps, using a 50×50 mesh, took approximately 35 to 40 minutes and outputted a solution vector of size $10,000 \times 1500$. The individual solution vectors for each of the components could then be extracted from the full solution vector and visualised using MATLAB's inbuilt *'surf'* function. Various simulations were run using different initial profiles for each of the components and different boundary flux conditions. The simulations deemed most realistic used the following initial conditions: initially, small patches of damaged myelin and full healthy myelin everywhere else, and no T-cells or chemokine present in the domain. T-cells were sources entirely from a constant flux around the entire domain and chemokine was sourced entirely from secretion from lesions. Furthermore, a mesh size of 50×50 was chosen for simulations as it produced a high definition output visualisation, although with a substantial runtime trade-off (see Figure 3).

(a) 20×20 Mesh (5.74 seconds)   (b) 50×50 Mesh (116.35 seconds)
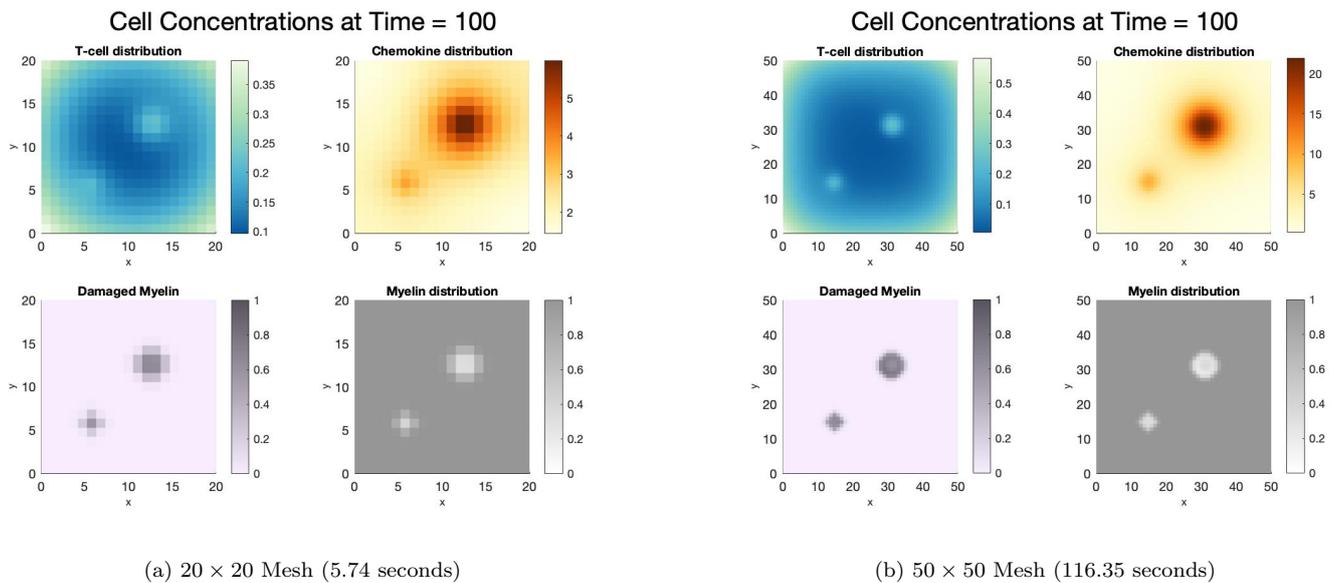
Figure 3: Efficiency and Fidelity Comparison with Change in Mesh Size

6   Findings

The main aspect of interest was the potential for different patterns and steady-state profiles emerging when testing different initial lesion profiles. Three different initial profiles we investigated and the simulation results are exhibited in Appendices D-F. The baseline simulation (see Appendix D) began with an initial lesion profile of a single small patch of damaged myelin in the centre of the domain. The first alternative simulation (see

Appendix E) began with two lesions, one smaller and one larger. The second alternative simulation (see Appendix F) began with one larger lesion in the centre surrounded evenly by four smaller lesions. The key finding from these simulations was that, regardless of the initial lesion profile, all simulations converged to the same steady-state profile of full myelin regeneration. This indicated that although simulations exhibited expected results, changes would have to be made in order to improve the realistic nature of the model and to truly capture the behaviour of MS lesions, as full myelin regeneration would never be completely successful in MS patients. In theory, not only would the T-cells be unable to regenerate the healthy myelin completely but they would tend to cause further damage and lesion growth.

In all of the simulations we can observe the expected dynamics of the model. Up until 150 timesteps we can see the initial lesion growing slowly and once the T-cells from the boundary chemotax towards the chemokine we begin to see the lesion eroding (around 250 timesteps). As the chemokine gradient is highest towards the centre of the lesion this is the location that begin to erode first, and once about 550 timesteps have passed the remnants of the lesion are barely visible. After the lesion is destroyed no more chemokine is secreted into the domain and it then diffuses and decays slowly. Similarly with the T-cells, once the lesion is destroyed they will still continue to travel into the domain, but over time will diffuse to a homogeneous steady-state.

Another metric we wanted to observe was the change in total volumes of each of the components over time, obtained by integrating over the domain. Figure 4 exhibits each of the components integrated across the domain over time, for 10 various parameter changes. Firstly, examining Figure 4a, we can see that all simulation garner the same result except for the simulation that substantially increased the T-cell decay rate (δ_T), as expected. Furthermore, in Figure 4b, all simulations exhibit the same general pattern for chemokine volume over time; the volume initially exhibits a rapid increase as the lesion grows and once the lesion is destroyed, a gradual decrease is exhibited as the chemokine decays. We can see the simulation that increased the chemokine decay rate (δ_C) exhibits a steeper decrease after lesion elimination, as expected. We can further observe the simulation that significantly increases the diffusion constant of the damaged myelin (D_m), has a great impact on the speed of lesion growth, and hence, chemokine secretion. Additionally, we can clearly observe in Figure 4c and 4d, the lesion growth and subsequent myelin regeneration mirrored in both figures. Independent of parameter values, we can again observe all simulations reaching the same steady-state of complete lesion elimination over time.

7 Conclusion

Overall, the objective to formulate a model that would capture the immune response triggered by Multiple Sclerosis demyelination and subsequently solve this system was achieved. The system was simulated numerically and visualised in MATLAB and outputted relevant and meaningful results that allowed us to gather important insights. Although the simulations and model were a success and exhibited expected and reasonably realistic preliminary results, there is much room for extension and improvement. A key objective for future studies of

(a) T-cell Volume over Time

(b) Chemokine Volume over Time

(c) Damaged Myelin Volume over Time

(d) Healthy Myelin Volume over Time
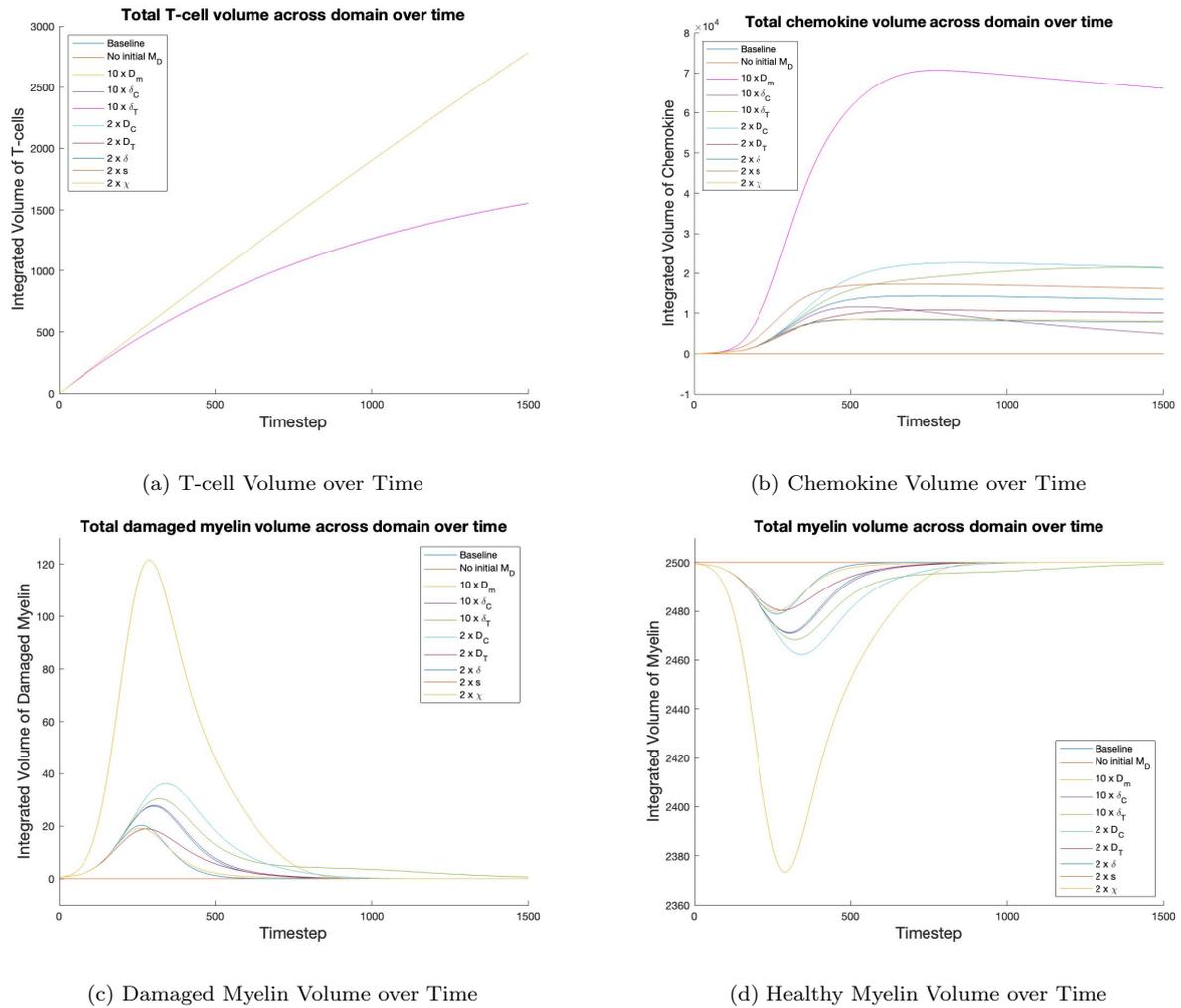
Figure 4: Integrated Volumes over Time

this model would be to either parameterise or non-dimensionalise the system to garner more meaningful results. A way in which the parameters could be estimated would involve further research into known cell behaviours (eg. rates of decay) and potentially, to calibrate the model with MS patient MRI data. This would certainly go a long way in improving the model to exhibit realistic results. Additionally, an evident limitation of the model was the simplified T-cell model. This was clear as the model captured T-cells only healing damaged myelin rather than mistakenly causing further damage to the myelin, as exhibited in MS patients. To capture this, a possible extension would be to include two different T-cell PDEs into the model, one which heals lesions and one which causes further demyelination. Another improvement to the T-cell model would be to make the flux of T-cells into the domain dependent on total volume of damaged myelin present. Finally, as exhibited in real MS patients, the model could further be adapted to capture the potential for random lesion generation in different areas of the brain.

14

References

[1] Dendrou C.A., Fugger L., Friese M.A. 2015, 'Immunopathology of multiple sclerosis', *Nature Reviews Immunology*, vol. 15, no. 9, pp. 545-58. https://doi.org/10.1038/nri3871

[2] Høglund, R.A. & Maghazachi, A.A. 2014, 'Multiple sclerosis and the role of immune cells', *World Journal of Experimental Medicine*, vol. 4, no. 3, pp. 27-37. DOI: 10.5493/wjem.v4.i3.27

[3] Hillen, T & Painter, K.J. 2009, 'A user's guide to PDE models for chemotaxis', *Journal of Mathematical Biology*, vol. 58, no. 183. https://doi.org/10.1007/s00285-008-0201-3

[4] Hemmer B., Kerschensteiner M., Korn T. 2015, 'Role of the innate and adaptive immune responses in the course of multiple sclerosis', *The Lancet Neurology*, vol. 14, no. 4, pp. 406-419. DOI: 10.1016/S1474-4422(14)70305-9

[5] Barresi, R., Bilotta, E., Gargano, F. et al. 2016, 'Wavefront invasion for a chemotaxis model of Multiple Sclerosis.', *Ricerche di Matematica*, vol. 65, pp. 423–434. https://doi.org/10.1007/s11587-016-0265-0

[6] Kotelnikova E., Kiani N.A., Abad E., Martinez-Lapiscina E.H., Andorra M., Zubizarreta I., Pulido-Valdeolivas I., Pertsovskaya I., Alexopoulos L.G., Olsson T., Martin R., Paul F., Tegnér J., Garcia-Ojalvo J., Villoslada P. 2017, 'Dynamics and heterogeneity of brain damage in multiple sclerosis', *PLOS Computational Biology*, vol. 13, no. 10. https://doi.org/10.1371/journal.pcbi.1005757

[7] Li X., Gruosso T., Zuo D., Omeroglu A., Meterissian S., Guiot M.C., Salazar A., Park M., Levine H. 2019, 'Infiltration of CD8+ T cells into tumor cell clusters in triple-negative breast cancer', *Proceedings of the National Academy of Sciences of the United States of America*, vol. 116, no. 9, pp. 3678-3687. https://doi.org/10.1073/pnas.1817652116

Appendix A: Chemokine Boundary Conditions

See below full set of boundary Finite Volume Equations for the chemokine PDE (Equation 9).

North Boundary Nodes:

$$\left[\frac{\partial C}{\partial y}\right]_n = 0, \qquad \therefore J_n = 0$$

Finite Volume Equation:

$$0 = \frac{dC_P(t)}{dt} + \frac{1}{\Delta x_P}(J_e - J_w) - \frac{J_s}{\Delta y_P} + \delta_C C_P - sM_{DP}$$

North-East Boundary Node:

$$\left[\frac{\partial C}{\partial y}\right]_n = \left[\frac{\partial C}{\partial x}\right]_e = 0, \qquad \therefore J_n = J_e = 0$$

Finite Volume Equation:

$$0 = \frac{dC_P(t)}{dt} - \frac{J_w}{\Delta x_P} - \frac{J_s}{\Delta y_P} + \delta_C C_P - sM_{DP}$$

East Boundary Nodes:

$$\left[\frac{\partial C}{\partial x}\right]_e = 0, \qquad \therefore J_e = 0$$

Finite Volume Equation:

$$0 = \frac{dC_P(t)}{dt} - \frac{J_w}{\Delta x_P} + \frac{1}{\Delta y_P}(J_n - J_s) + \delta_C C_P - sM_{DP}$$

South-East Boundary Node:

$$\left[\frac{\partial C}{\partial y}\right]_s = \left[\frac{\partial C}{\partial x}\right]_e = 0, \qquad \therefore J_s = J_e = 0$$

Finite Volume Equation:

$$0 = \frac{dC_P(t)}{dt} - \frac{J_w}{\Delta x_P} + \frac{J_n}{\Delta y_P} + \delta_C C_P - sM_{DP}$$

South Boundary Nodes:

$$\left[\frac{\partial C}{\partial y}\right]_s = 0, \qquad \therefore J_s = 0$$

Finite Volume Equation:

$$0 = \frac{dC_P(t)}{dt} + \frac{1}{\Delta x_P}(J_e - J_w) + \frac{J_n}{\Delta y_P} + \delta_C C_P - sM_{DP}$$

South-West Boundary Node:

$$\left[\frac{\partial C}{\partial y}\right]_s = \left[\frac{\partial C}{\partial x}\right]_w = 0, \qquad \therefore J_s = J_w = 0$$

Finite Volume Equation:

$$0 = \frac{dC_P(t)}{dt} + \frac{J_e}{\Delta x_P} + \frac{J_n}{\Delta y_P} + \delta_C C_P - sM_{DP}$$

West Boundary Nodes:
$$\left[\frac{\partial C}{\partial x}\right]_w = 0, \qquad \therefore J_w = 0$$

Finite Volume Equation:

$$0 = \frac{dC_P(t)}{dt} + \frac{J_e}{\Delta x_P} + \frac{1}{\Delta y_P}(J_n - J_s) + \delta_C C_P - s M_{DP}$$

North-West Boundary Node:
$$\left[\frac{\partial C}{\partial y}\right]_n = \left[\frac{\partial C}{\partial x}\right]_w = 0, \qquad \therefore J_n = J_w = 0$$

Finite Volume Equation:

$$0 = \frac{dC_P(t)}{dt} + \frac{J_e}{\Delta x_P} - \frac{J_s}{\Delta y_P} + \delta_C C_P - s M_{DP}$$

Appendix B: T-cell Boundary Conditions

See below full set of boundary Finite Volume Equations for the T-cell PDE (Equation 10).

North Boundary Nodes:
$$\left[\frac{\partial C}{\partial y}\right]_n = \left[\frac{\partial T}{\partial y}\right]_n = 0, \qquad \therefore Q_n = 0$$

Finite Volume Equation:

$$0 = \frac{dT_P(t)}{dt} + \frac{1}{\Delta x_P}(Q_e - Q_w) - \frac{Q_s}{\Delta y_P} + \delta_T T_P$$

North-East Boundary Node:
$$\left[\frac{\partial C}{\partial y}\right]_n = \left[\frac{\partial C}{\partial x}\right]_e = \left[\frac{\partial T}{\partial y}\right]_n = \left[\frac{\partial T}{\partial x}\right]_e = 0, \qquad \therefore Q_n = Q_e = 0$$

Finite Volume Equation:
$$0 = \frac{dT_P(t)}{dt} - \frac{Q_w}{\Delta x_P} - \frac{Q_s}{\Delta y_P} + \delta_T T_P$$

East Boundary Nodes:
$$\left[\frac{\partial C}{\partial x}\right]_e = \left[\frac{\partial T}{\partial x}\right]_e = 0, \qquad \therefore Q_e = 0$$

Finite Volume Equation:

$$0 = \frac{dT_P(t)}{dt} - \frac{Q_w}{\Delta x_P} + \frac{1}{\Delta y_P}(Q_n - Q_s) + \delta_T T_P$$

South-East Boundary Node:
$$\left[\frac{\partial C}{\partial y}\right]_s = \left[\frac{\partial C}{\partial x}\right]_e = \left[\frac{\partial T}{\partial y}\right]_s = \left[\frac{\partial T}{\partial x}\right]_e = 0, \qquad \therefore Q_s = Q_e = 0$$

Finite Volume Equation:
$$0 = \frac{dT_P(t)}{dt} - \frac{Q_w}{\Delta x_P} + \frac{Q_n}{\Delta y_P} + \delta_T T_P$$

South Boundary Nodes:

$$\left[\frac{\partial C}{\partial y}\right]_s = \left[\frac{\partial T}{\partial y}\right]_s = 0, \qquad \therefore Q_s = 0$$

Finite Volume Equation:

$$0 = \frac{dT_P(t)}{dt} + \frac{1}{\Delta x_P}(Q_e - Q_w) + \frac{Q_n}{\Delta y_P} + \delta_T T_P$$

South-West Boundary Node:

$$\left[\frac{\partial C}{\partial y}\right]_s = \left[\frac{\partial C}{\partial x}\right]_w = \left[\frac{\partial T}{\partial y}\right]_s = \left[\frac{\partial T}{\partial x}\right]_w = 0, \qquad \therefore Q_s = Q_w = 0$$

Finite Volume Equation:

$$0 = \frac{dT_P(t)}{dt} + \frac{Q_e}{\Delta x_P} + \frac{Q_n}{\Delta y_P} + \delta_T T_P$$

West Boundary Nodes:

$$\left[\frac{\partial C}{\partial x}\right]_w = \left[\frac{\partial T}{\partial x}\right]_w = 0, \qquad \therefore Q_w = 0$$

Finite Volume Equation:

$$0 = \frac{dT_P(t)}{dt} + \frac{Q_e}{\Delta x_P} + \frac{1}{\Delta y_P}(Q_n - Q_s) + \delta_T T_P$$

North-West Boundary Node:

$$\left[\frac{\partial C}{\partial y}\right]_n = \left[\frac{\partial C}{\partial x}\right]_w = \left[\frac{\partial T}{\partial y}\right]_n = \left[\frac{\partial T}{\partial x}\right]_w = 0, \qquad \therefore Q_n = Q_w = 0$$

Finite Volume Equation:

$$0 = \frac{dT_P(t)}{dt} + \frac{Q_e}{\Delta x_P} - \frac{Q_s}{\Delta y_P} + \delta_T T_P$$

Appendix C: Full Spatial Discretisation of Damaged Myelin PDE

Following the same derivation method of the FVE used in section 4.2.1, we take Equation 8 and integrate over the control volume, such that:

$$\frac{\partial M_D(x,t)}{\partial t} = D_M \nabla^2 M_D + vMM_D - \delta M_D T$$

$$0 = \int\int_{V_p}\frac{\partial M_D(x,t)}{\partial t}dV + \int\int_{V_p}\nabla \cdot \mathbf{W}\,dV - v\int\int_{V_p}MM_D\,dV + \delta\int\int_{V_p}M_D T\,dV \qquad (29)$$

Where,

$$\mathbf{W} = -D_M\nabla M_D(x,t) \qquad (30)$$

Using the relevant control volume averaged values of M_D, M, T and $\frac{\partial M_D}{\partial t}$ over V_p, this can be simplified:

$$0 = \frac{d\bar{M}_{DP}(t)}{dt} + \frac{1}{\Delta x_P \Delta y_P}\int\int_{V_p}\nabla \cdot \mathbf{W}\,dV - v\bar{M}_P\bar{M}_{DP} - \delta\bar{M}_{DP}\bar{T}_P$$

Again using Green's Theorem to evaluate the second double integral, we arrive at the following approximation:

$$\int\int_{V_p}\nabla \cdot \mathbf{W}\,dV \approx \Delta y_P(W_e - W_w) + \Delta x_P(W_n - W_s)$$

Thus,

$$0 = \frac{d\bar{M}_{DP}(t)}{dt} + \frac{1}{\Delta x_P}(W_e - W_w) + \frac{1}{\Delta y_P}(W_n - W_s) - v\bar{M}_P\bar{M}_{DP} - \delta\bar{M}_{DP}\bar{T}_P \tag{31}$$

Using Equation 30, a standard forward difference approximation for the partial derivatives and an average approximation of our M_D terms, we can now evaluate our W terms.

$$W_e = -D_m\left[\frac{\partial M_D}{\partial x}\right]_e \approx -D_m\left(\frac{M_{DE} - M_{DP}}{\delta x_e}\right)$$

$$W_w = -D_m\left[\frac{\partial M_D}{\partial x}\right]_w \approx -D_m\left(\frac{M_{DP} - M_{DW}}{\delta x_w}\right)$$

$$W_n = -D_m\left[\frac{\partial M_D}{\partial y}\right]_n \approx -D_m\left(\frac{M_{DN} - M_{DP}}{\delta y_n}\right)$$

$$W_s = -D_m\left[\frac{\partial M_D}{\partial y}\right]_s \approx -D_m\left(\frac{M_{DP} - M_{DS}}{\delta y_s}\right)$$

Where M_{DP} is the current concentration of damaged myelin at our particular node, M_{DE} is the concentration at the node to the right of our particular node, M_{DW} is the concentration at the node to the left, M_{DN} is the concentration at the node above and M_{DS} is the concentration at the node below (see Figure 2 for full visualisation of nodal design).

Hence, substituting our W values into Equation 31 and letting $M_{DP} \approx \bar{M}_{DP}$, $M_P \approx \bar{M}_P$ and $T_P \approx \bar{T}_P$ we can derive our full Finite Volume Equation:

$$0 = \frac{dM_{DP}(t)}{dt} + \frac{D_m}{\Delta x_P}\left(\frac{M_{DP} - M_{DW}}{\delta x_w} - \frac{M_{DE} - M_{DP}}{\delta x_e}\right) + \frac{D_m}{\Delta y_P}\left(\frac{M_{DP} - M_{DS}}{\delta y_s} - \frac{M_{DN} - M_{DP}}{\delta y_n}\right) - vM_PM_{DP} - \delta M_{DP}T_P$$

Boundary Conditions:

For this damaged myelin PDE we assume no-flux boundary conditions along all sides of the domain (see Equation 12) we can derive alternative Finite Volume Equations (adapted from Equation 23) accordingly for each boundary node.

See below full set of boundary Finite Volume Equations for the damaged myelin PDE:

North Boundary Nodes:

$$\left[\frac{\partial M_D}{\partial y}\right]_n = 0, \qquad \therefore W_n = 0$$

Finite Volume Equation:

$$0 = \frac{dM_{DP}(t)}{dt} + \frac{1}{\Delta x_P}(W_e - W_w) - \frac{W_s}{\Delta y_P} - vM_PM_{DP} - \delta M_{DP}T_P$$

North-East Boundary Node:

$$\left[\frac{\partial M_D}{\partial y}\right]_n = \left[\frac{\partial M_D}{\partial x}\right]_e = 0, \qquad \therefore W_n = W_e = 0$$

Finite Volume Equation:

$$0 = \frac{dM_{DP}(t)}{dt} - \frac{W_w}{\Delta x_P} - \frac{W_s}{\Delta y_P} - vM_P M_{DP} - \delta M_{DP} T_P$$

East Boundary Nodes:

$$\left[\frac{\partial M_D}{\partial x}\right]_e = 0, \qquad \therefore W_e = 0$$

Finite Volume Equation:

$$0 = \frac{dM_{DP}(t)}{dt} - \frac{W_w}{\Delta x_P} + \frac{1}{\Delta y_P}(W_n - W_s) - vM_P M_{DP} - \delta M_{DP} T_P$$

South-East Boundary Node:

$$\left[\frac{\partial M_D}{\partial y}\right]_s = \left[\frac{\partial M_D}{\partial x}\right]_e = 0, \qquad \therefore W_s = W_e = 0$$

Finite Volume Equation:

$$0 = \frac{dM_{DP}(t)}{dt} - \frac{W_w}{\Delta x_P} + \frac{W_n}{\Delta y_P} - vM_P M_{DP} - \delta M_{DP} T_P$$

South Boundary Nodes:

$$\left[\frac{\partial M_D}{\partial y}\right]_s = 0, \qquad \therefore W_s = 0$$

Finite Volume Equation:

$$0 = \frac{dM_{DP}(t)}{dt} + \frac{1}{\Delta x_P}(W_e - W_w) + \frac{W_n}{\Delta y_P} - vM_P M_{DP} - \delta M_{DP} T_P$$

South-West Boundary Node:

$$\left[\frac{\partial M_D}{\partial y}\right]_s = \left[\frac{\partial M_D}{\partial x}\right]_w = 0, \qquad \therefore W_s = W_w = 0$$

Finite Volume Equation:

$$0 = \frac{dM_{DP}(t)}{dt} + \frac{W_e}{\Delta x_P} + \frac{W_n}{\Delta y_P} - vM_P M_{DP} - \delta M_{DP} T_P$$

West Boundary Nodes:

$$\left[\frac{\partial M_D}{\partial x}\right]_w = 0, \qquad \therefore W_w = 0$$

Finite Volume Equation:

$$0 = \frac{dM_{DP}(t)}{dt} + \frac{W_e}{\Delta x_P} + \frac{1}{\Delta y_P}(W_n - W_s) - vM_P M_{DP} - \delta M_{DP} T_P$$

North-West Boundary Node:

$$\left[\frac{\partial M_D}{\partial y}\right]_n = \left[\frac{\partial M_D}{\partial x}\right]_w = 0, \qquad \therefore W_n = W_w = 0$$

Finite Volume Equation:

$$0 = \frac{dM_{DP}(t)}{dt} + \frac{W_e}{\Delta x_P} - \frac{W_s}{\Delta y_P} - vM_P M_{DP} - \delta M_{DP} T_P$$

Appendix D: Baseline Simulation Visualisation
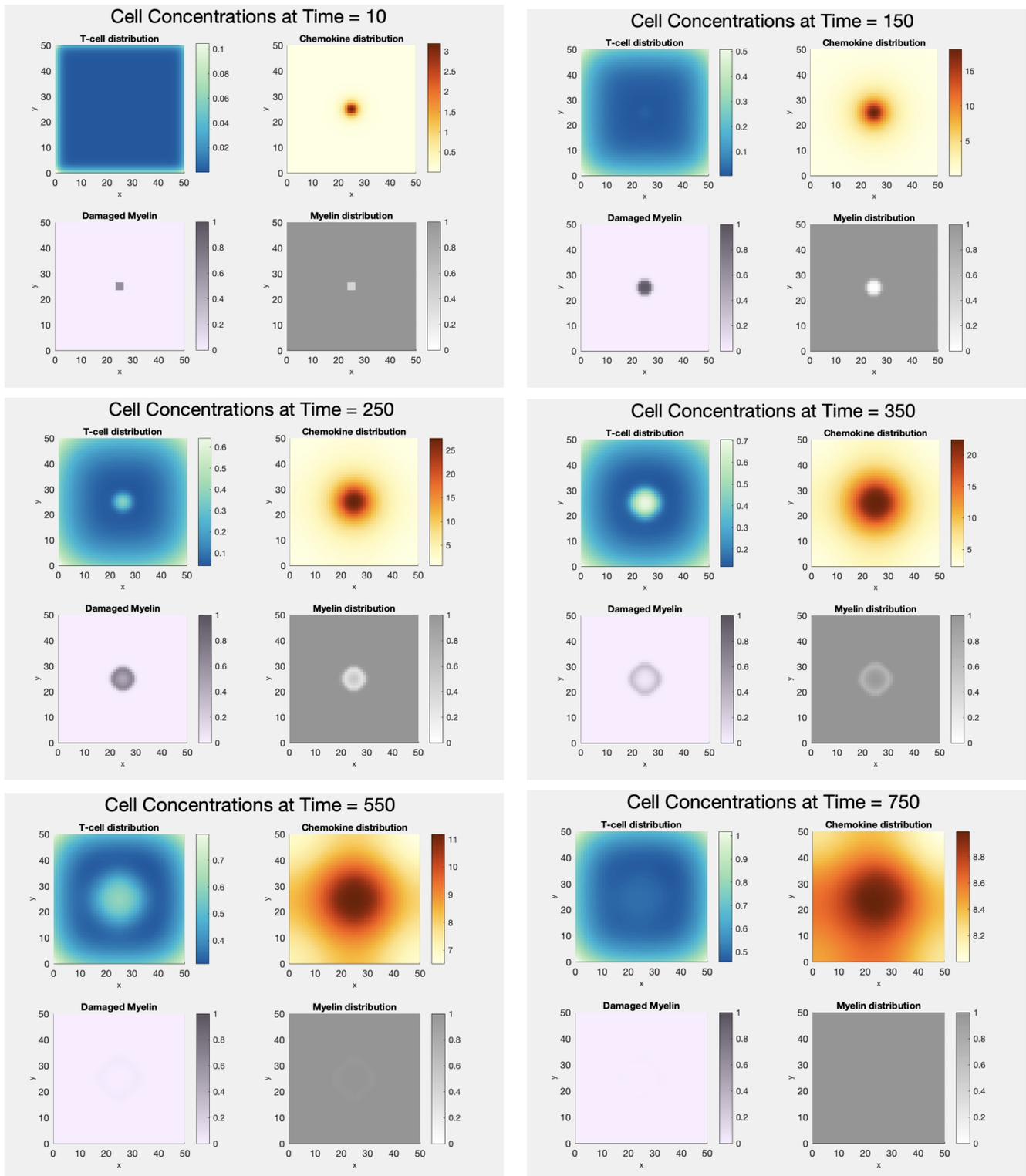
Figure 5: Baseline Simulation Results Visualised over Time

Appendix E: First Alternative Initial Profile Simulation Visualisation
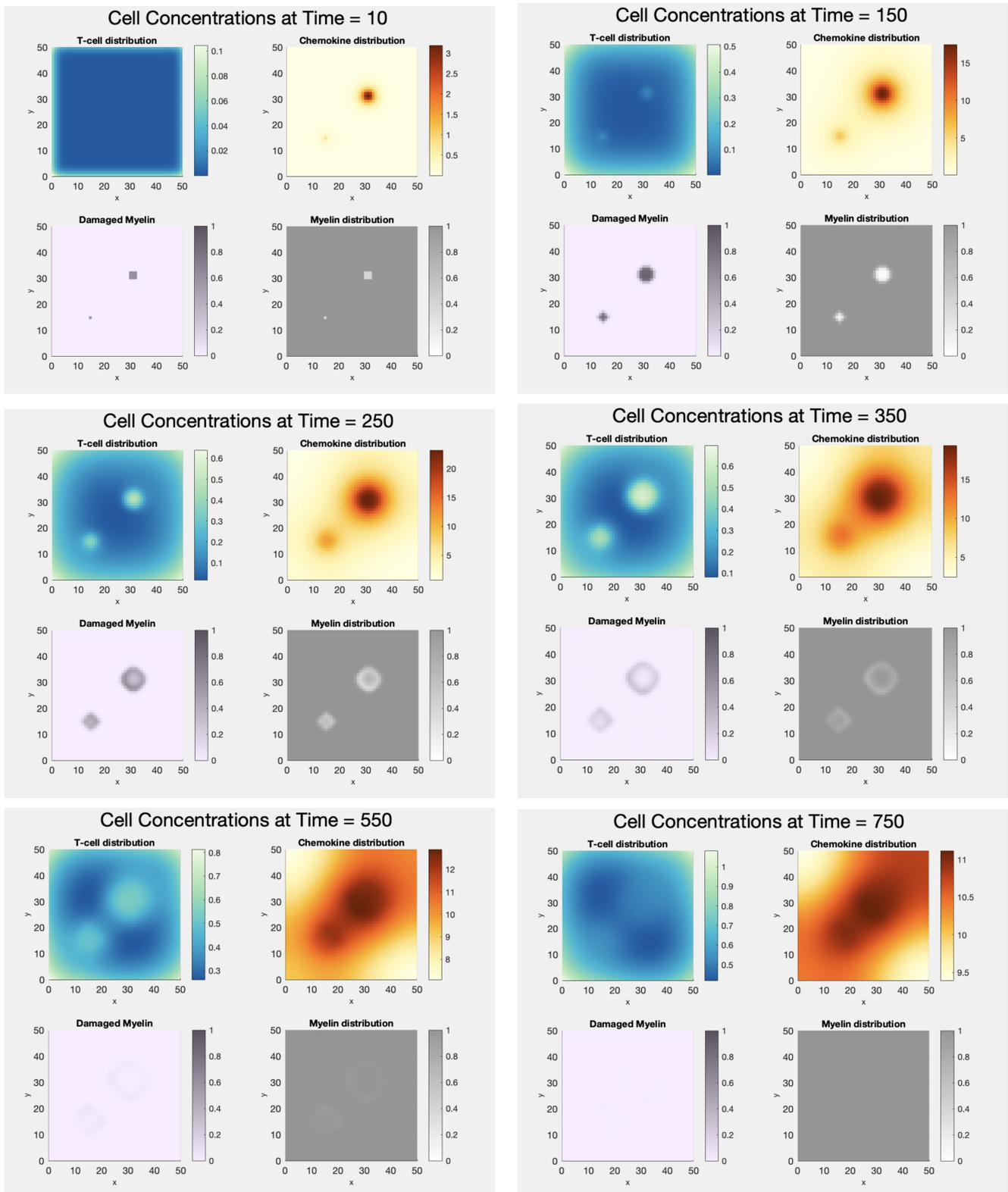
Figure 6: First Alternative Simulation Results Visualised over Time

Appendix F: Second Alternative Initial Profile Simulation Visualisation
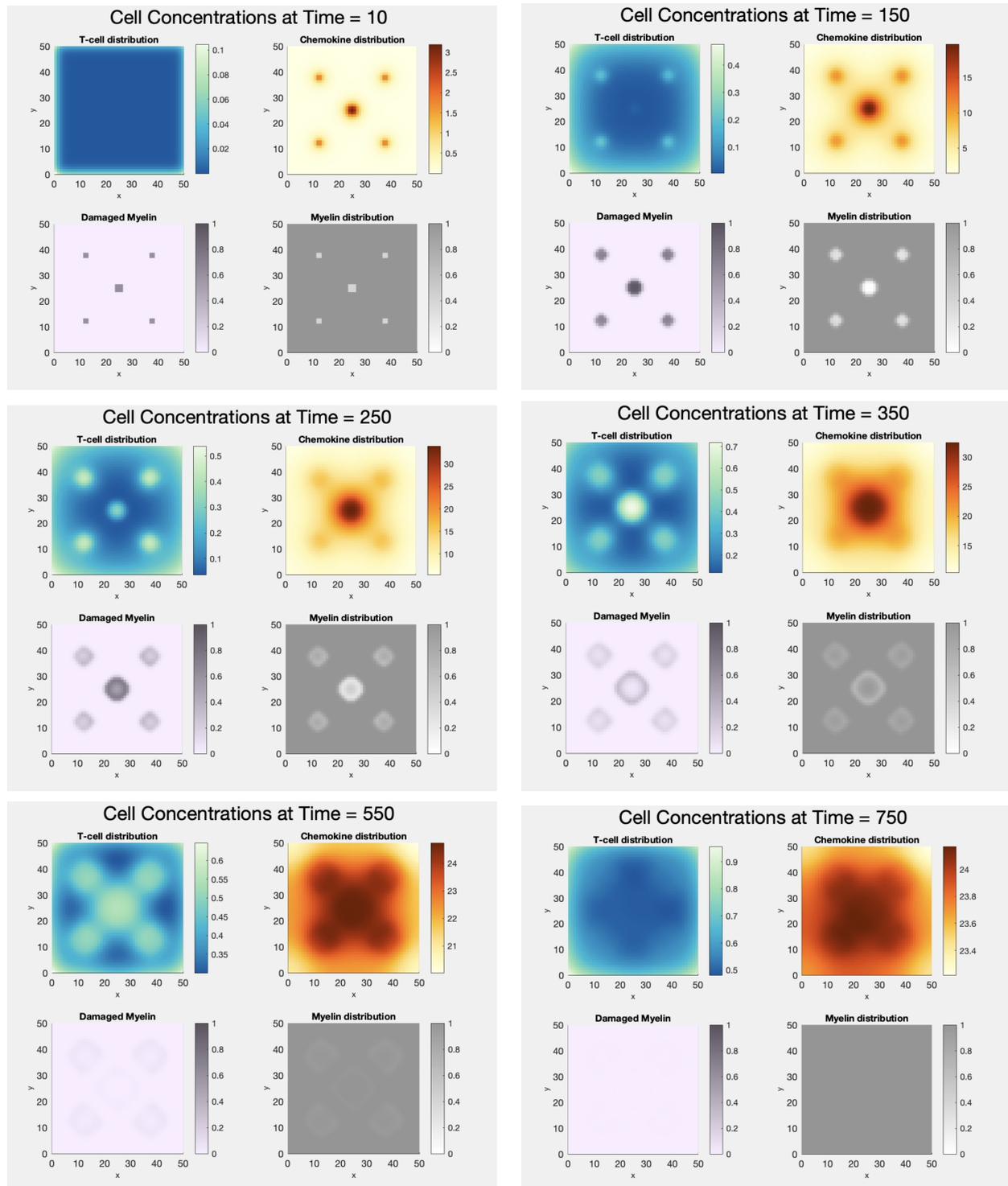
Figure 7: Second Alternative Simulation Results Visualised over Time