

Numerical Methods in Magnetic Resonance Imaging: The Bloch-Torrey Equation

Jonathan Doucette^{a,b}

^aUBC MRI Research Centre, University of British Columbia, 2221 Wesbrook Mall, Vancouver, BC, Canada.

^bDepartment of Physics and Astronomy, University of British Columbia, 6224 Agricultural Road, Vancouver, BC, Canada.

April 12, 2018

TODO: final spacing In magnetic resonance imaging, insight into different imaging modalities can be gained through the simulation of the magnetic resonance signal measured by the scanner. This signal is modelled as the solution of the parabolic Bloch-Torrey partial differential equation. In this work, I compare various numerical techniques for solving the Bloch-Torrey equation in the presence of highly discontinuous data with the goal of optimising the trade-off between solution accuracy and computation time.

Introduction

In magnetic resonance imaging (MRI), insight into different imaging modalities can be gained through the simulation of the magnetic resonance (MR) signal from first principles. The MR signal which is measured by an MRI scanner within a given region is directly proportional to the magnitude of the net magnetization vector \mathbf{M} within that region. In tissue, \mathbf{M} arises due to the superposition of the magnetic moments of water molecules, also known as *spins* in MRI nomenclature. Ordinarily, \mathbf{M} is zero in tissue as spins are randomly oriented and therefore their vector sum is zero on the average. In MRI, however, there is a large and constant external magnetic field $\mathbf{B}_0 = B_0 \hat{\mathbf{z}}$ which forces the alignment of the magnetized spins with \mathbf{B}_0 , taken here to be along the z -direction. This allows for the measurement, and importantly, the manipulation of the otherwise negligible net magnetization \mathbf{M} of the spins.

In one common type of MRI scan, a *gradient echo* scan, \mathbf{M} is initially flipped into the xy -plane through the application of a radio frequency (RF) magnetic pulse. Immediately, \mathbf{M} begins to realign with \mathbf{B}_0 exponentially quickly. The rate $R(\mathbf{x})$ at which this realignment occurs, however, depends on local tissue properties. Additionally, there is a characteristic precession of \mathbf{M} about \mathbf{B}_0 as it realigns. This precession occurs at a rate $\omega(\mathbf{x})$ which, too, depends on the magnetic properties of the environment. Lastly, spins are free to diffuse within their environment while realigning, which means that they will see varying $R(\mathbf{x})$ and $\omega(\mathbf{x})$ as they move through the tissue.

The Bloch-Torrey equation

In the continuum limit, the net magnetization \mathbf{M} within a given region is modelled as a continuous vector field, and the complex dynamics which follow the initial RF pulse can be beautifully modelled as solutions to a parabolic partial differential equation (PDE) called the Bloch-Torrey equation [1]:

$$\begin{cases} u_t = D\Delta u - Ru + \omega v \\ v_t = D\Delta v - Rv - \omega u \end{cases} \quad (1)$$

Here, u and v are the x - and y -components of the magnetization \mathbf{M} and D is the diffusion constant. For short simulation times, or *echo times*, T , the z -component of \mathbf{M} does not change appreciably, and we need only consider the transverse magnetization $\mathbf{M}_\perp = (u, v)$.

The transverse Bloch-Torrey equation (1) may be equivalently written as

$$\mathcal{M}_t = D\Delta \mathcal{M} - \Gamma \mathcal{M} \quad (2)$$

where $\mathcal{M} = u + iv$ is the complex magnetization and $\Gamma(\mathbf{x}) = R(\mathbf{x}) + i\omega(\mathbf{x})$ is the complex decay rate. This form is both notationally convenient and conceptually illustrative; now, realignment with \mathbf{B}_0 corresponds to the magnitude of the (complex) transverse magnetization \mathcal{M} decaying to zero, and the precession of \mathbf{M} about \mathbf{B}_0 corresponds to the rotation of \mathcal{M} about the origin in the complex plane.

Signal simulation

This work will compare three different methods of simulating the signal $S(T)$ measured by an MRI scanner at a time T following an initial RF pulse: finite differences, splitting methods, and finite element methods.

The MR signal $S(T)$ is given by

$$S(T) = \left\| \int_{\Omega} \mathbf{M}_\perp(T) \, d\mathbf{x} \right\| \quad (3)$$

where $\mathbf{M}_\perp(T)$ is computed by solving (1) or (2). The initial transverse magnetization $\mathbf{M}_\perp(t = 0)$ will be taken to be $[0, 1]^T$ without loss of generality.

e-mail: jdoucette@phas.ubc.ca

Geometry

The transverse magnetization \mathbf{M}_\perp will be simulated within a cubic domain Ω of size $3 \times 3 \times 3 \text{ mm}^3$ with a single cylinder of a variable radius a in the centre of the domain, as pictured in Figure 1.

This geometry is chosen for two reasons. First, biologically it represents a single vessel present within a cubic imaging voxel. Second, there exists an exact solution for $\omega(\mathbf{x})$ given the magnetic susceptibility χ of the cylindrical blood vessel [2]. For a cylinder of radius a , we have that

$$\omega(\mathbf{x}) = \begin{cases} \frac{\chi B_0}{2} \sin^2 \theta \frac{a^2}{x^2+y^2} \frac{y^2-x^2}{x^2+y^2}, & \text{outside cylinder} \\ \frac{\chi B_0}{6} (3 \cos^2 \theta - 1), & \text{inside cylinder} \end{cases} \quad (4)$$

where θ is the angle between the cylinder axis and \mathbf{B}_0 . An example cross section of $\omega(\mathbf{x})$ can be seen in Figure 1. Note that $\omega(\mathbf{x})$ is constant within the cylinder, and jumps discontinuously to a smooth solution outside of the cylinder.

The transverse relaxation rate $R(\mathbf{x})$ is taken to be piecewise constant values inside and outside of the cylinder,

$$R(\mathbf{x}) = \begin{cases} R_{\text{tissue}}, & \text{outside cylinder} \\ R_{\text{blood}}, & \text{inside cylinder.} \end{cases} \quad (5)$$

where $0 < R_{\text{tissue}} < R_{\text{blood}}$ are positive constants.

Properties of the Bloch-Torrey equation

Static solution

If $D = 0$, solutions to (2) are simply complex exponentials

$$\mathcal{M}(\mathbf{x}, t) = \mathcal{M}_0 e^{-\Gamma(\mathbf{x})t} = \left(e^{-R(\mathbf{x})t} |\mathcal{M}_0| \right) e^{i(\phi_0 - \omega(\mathbf{x})t)}, \quad (6)$$

where we have written $\mathcal{M}_0 = |\mathcal{M}_0| e^{i\phi_0}$ in polar form. Equivalently, the static solution may be written

$$\mathbf{M}_\perp = e^{-R(\mathbf{x})t} \|\mathbf{M}_{\perp 0}\| \begin{pmatrix} \cos(\phi_0 - \omega(\mathbf{x})t) \\ \sin(\phi_0 - \omega(\mathbf{x})t) \end{pmatrix}. \quad (7)$$

In this form, it is clear to see that the magnitude $|\mathcal{M}| = \|\mathbf{M}_\perp\|$ of the transverse magnetization is exponentially damped at rate $R(\mathbf{x})$ and that the initial phase ϕ_0 changes at the constant rate $\omega(\mathbf{x})$, which can be interpreted as the transverse magnetization rotating about \mathbf{B}_0 at a constant angular velocity at each point in space.

Relation to the heat equation

If $\Gamma = 0$, then (1) reduces to two uncoupled heat equations in the transverse magnetization components u and v . This is equivalent to the limit as the magnetic field $B_0 \rightarrow 0$, as the magnetic moments of the spins are not forced to realign with an external field and instead are free to diffuse unhindered.

Mathematically, we should expect that solutions to the Bloch-Torrey equation exhibit some of the same properties as solutions of the heat equation, such as exponential suppression of high frequency eigenmodes.

Relation to the reaction-diffusion equation

The two-component reaction-diffusion equation, which models for example how the concentrations u and v of two chemicals change over time within a medium, is given by

$$\begin{cases} u_t = D\Delta u + F(u, v) \\ v_t = D\Delta v + G(u, v) \end{cases} \quad (8)$$

where F and G govern local chemical reactions which may increase or decrease the concentrations u and v depending on whether they are being produced or consumed.

The reaction-diffusion equation (8) corresponds directly to the Bloch-Torrey equation in the case of linear reaction terms $F = -Ru + \omega v$ and $G = -Rv - \omega u$. In this interpretation, $R(\mathbf{x})$ represents the local rate at which u and v are reacted away into waste products, and $\omega(\mathbf{x})$ represents the local rate at which u reacts into v and vice versa. **TODO:**

double check this interpretation

Weak form

The coupled system of PDEs (1) can be written as

$$\begin{cases} \mathbf{u}_t = -A\mathbf{u} & \text{in } \Omega, \quad t > 0 \\ \mathbf{u} = \mathbf{g} & \text{in } \Omega, \quad t = 0 \end{cases} \quad (9)$$

where $\mathbf{u} = [u, v]^T$ and

$$A = \begin{pmatrix} -D\Delta + R & -\omega \\ \omega & -D\Delta + R \end{pmatrix}. \quad (10)$$

Now, consider the vector test function $\mathbf{v} = [w, z]^T$ where $w, z \in V = H^1$. Then, the weak form of (9) is given by

$$\langle \mathbf{v}, \mathbf{u}_t \rangle = -\langle \mathbf{v}, A\mathbf{u} \rangle, \quad \forall \mathbf{v} \in V \times V \quad (11)$$

where

$$\langle \mathbf{v}, A\mathbf{u} \rangle = \int_{\Omega} \{ -Dw\Delta u + Rwu - \omega wv - Dz\Delta v + Rzv + \omega zu \} \, dx.$$

Applying zero Neumann boundary conditions on the boundary $\partial\Omega$, this simplifies to

$$\langle \mathbf{v}, A\mathbf{u} \rangle = \int_{\Omega} D(\nabla w \cdot \nabla u + \nabla z \cdot \nabla v) + R(wu + zv) + \omega(zu - wv) \, dx. \quad (12)$$

Coercivity and local magnetization decay

The bilinear form $\langle \mathbf{v}, A\mathbf{u} \rangle$ is coercive, as

$$\begin{aligned} \langle \mathbf{u}, A\mathbf{u} \rangle &= \int_{\Omega} D (|\nabla u|^2 + |\nabla v|^2) + R(u^2 + v^2) dx \\ &\geq \left(\min_{\mathbf{x}} R(\mathbf{x}) \right) \|\mathbf{u}\|_{L^2}^2 \end{aligned} \quad (13)$$

with $\min_{\mathbf{x}} R(\mathbf{x}) = R_{\text{tissue}} > 0$. From this, it is then easy to show that the L^2 -norm of the solution decreases in time:

$$\langle \mathbf{u}, \mathbf{u}_t \rangle = \int_{\Omega} uu_t + vv_t dx = \frac{1}{2} \frac{\partial}{\partial t} \int_{\Omega} u^2 + v^2 dx = \frac{1}{2} \frac{\partial}{\partial t} \|\mathcal{M}\|_{L^2}^2,$$

and so, for non-zero \mathbf{u} , we have that

$$\frac{1}{2} \frac{\partial}{\partial t} \|\mathcal{M}(\mathbf{x}, t)\|_{L^2}^2 = \langle \mathbf{u}, \mathbf{u}_t \rangle = -\langle \mathbf{u}, A\mathbf{u} \rangle < 0$$

and therefore $\|\mathcal{M}\|_{L^2}^2$ decreases monotonically with time when starting from a non-zero \mathcal{M}_0 . Note that this does not imply that $S(t)$ decreases monotonically as well, although it does of course go to zero in the limit as $t \rightarrow \infty$.

Uniqueness

Suppose for contradiction that \mathbf{u}_1 and \mathbf{u}_2 are different solutions to (9) zero Neumann boundary conditions on $\partial\Omega$.

Then, let $\mathbf{w} = \mathbf{u}_1 - \mathbf{u}_2$. By linearity, we have that \mathbf{w} solves

$$\begin{cases} \mathbf{w}_t = -A\mathbf{w} & \text{in } \Omega, \quad t > 0 \\ \mathbf{w} = \mathbf{0} & \text{in } \Omega, \quad t = 0. \end{cases}$$

Now, we have that $\|\mathbf{w}(t=0)\|_{L^2} = 0$, and since we have shown that $\|\mathbf{u}\|_{L^2}$ is non-increasing for all solutions \mathbf{u} to (9), it must be that $\|\mathbf{w}\|_{L^2} \equiv 0$ for all time and therefore \mathbf{u}_1 must equal \mathbf{u}_2 almost everywhere for all time, a contradiction.

Existence and continuous dependence on data

Existence and continuous dependence on data of (weak) solutions to the Bloch-Torrey equation (1) follows directly from the analogy with the linear reaction-diffusion equations (8). Such results are greatly detailed in many texts, such as for example [3] and [4], and I will not go into further details here, except to say that (1) is indeed well posed with Neumann boundary conditions on $\partial\Omega$.

Methods

Finite difference methods

The simplest method for solving equation (2) is through a simple application of the method of lines to discretise the

negative Laplacian $-\Delta$ using standard second order consistent centred differences, resulting in $-\Delta^h$ such that

$$-\Delta^h \mathcal{M}(\mathbf{x}_i) = \frac{6\mathcal{M}(\mathbf{x}_i) - \sum_{j \in \mathcal{J}_i} \mathcal{M}(\mathbf{x}_j)}{h^2} + \mathcal{O}(h^2) \quad (14)$$

where h is the uniform grid spacing and the sum is over the six neighbours of \mathbf{x}_i indexed by \mathcal{J}_i . It should be noted that this second order consistency will not actually be observed in practice, as due to the discontinuities in $R(\mathbf{x})$ and $\omega(\mathbf{x})$ we expect that the solution may have jumps in the gradient on the cylinder boundary.

Boundary conditions are taken to be periodic on the boundaries of the unit cube [5], as opposed to Neumann boundary conditions on the boundary of the cube and cylinder. This is due to simplicity of implementation and the inherent limitations of finite differences methods on irregular domains. Additionally, Γ is discretised to the diagonal matrix operator Γ^h such that $\Gamma_{ii}^h = \Gamma(\mathbf{x}_i)$, where \mathbf{x}_i is a point on the discretised grid. What results is the discrete system

$$\frac{\partial}{\partial t} \mathcal{M}^h(t) = -A^h \mathcal{M}^h(t) \quad (15)$$

$$\Rightarrow \mathcal{M}^h(t) = e^{-A^h t} \mathcal{M}^h(0), \quad (16)$$

where $A^h = -D\Delta^h + \Gamma^h$ and the notation $e^{-A^h t}$ represents the *matrix exponential* [6] of the matrix $-A^h t$. In our case, although $A^h \in \mathbb{C}^{N^3 \times N^3}$ where N may be large, we need only compute the matrix exponential vector product – also known as the *action* of the matrix exponential – and not $e^{-A^h t}$ itself, which is in general full despite A^h being sparse [6]. Algorithms for computing the action of the matrix exponential are known [7, 8]. I use the algorithm described by Higham et al. in [9], as well as their MATLAB code *expmv*.

Note that although the periodic Laplacian operator Δ^h is singular due to the fact that the constant solution has eigenvalue zero, adding Γ^h shifts the positive constant diagonal $6/h^2$ up by $R(\mathbf{x}_i) + i\omega(\mathbf{x}_i)$, thereby making the matrix A^h strongly diagonally dominant and therefore non-singular by the Gershgorin circle theorem [10]. Therefore, the solution (16) to the system (15) exists and is unique.

The resulting signal $S^h(T)$ is computed by integrating the discrete solution $\mathcal{M}^h(T) = e^{-A^h T} \mathcal{M}_0^h$ using the trapezoidal rule and taking the absolute value of the result.

Operator splitting methods

Operator splitting methods for parabolic PDEs are useful when a differential operator has no closed form operator exponential, but can be split into a sum of operators which do. The operator exponential is the formal equivalent of the matrix exponential for differential operators, namely it is the operator defined such that $\mathbf{u}(\mathbf{x}, t) = e^{-At} \mathbf{u}_0(\mathbf{x})$, where $\partial_t \mathbf{u}(\mathbf{x}) = -A\mathbf{u}(\mathbf{x})$ with $\mathbf{u}_0(\mathbf{x})$ given.

Suppose a differential operator $A = T + V$ is given such that the action of the operator exponential $e^{-At} \mathbf{u}$ has no closed

form, but the actions $e^{-Tt}\mathbf{u}$ and $e^{-Vt}\mathbf{u}$ do. Then, one can use the so-called *Strang splitting* [11, 12] to approximate the action of the operator exponential of A with that of T and V :

$$e^{-A\delta t} = e^{-(T+V)\delta t} \quad (17)$$

$$= e^{-T\delta t} e^{-V\delta t} + \mathcal{O}(\delta t^2) \quad (18)$$

$$= e^{-V\delta t/2} e^{-T\delta t} e^{-V\delta t/2} + \mathcal{O}(\delta t^3) \quad (19)$$

where the approximation errors are incurred due to the fact that T and V are assumed to not commute, i.e. $T(V\mathbf{u}) \neq V(T\mathbf{u})$. Were they to commute, both of these approximations would be exact.

In our case, from equation (2) we may take $A = T+V$ where $T = -D\Delta$ with periodic boundary conditions, and $V = \Gamma$. Then, the action of the evolution operators $e^{Dt\Delta}$ and $e^{-\Gamma t}$ in approximations (18) and (19) are known [13]:

$$e^{Dt\Delta}\mathcal{M}_0(\mathbf{x}) = \Phi(\mathbf{x}, t) * \mathcal{M}_0(\mathbf{x}) \quad (20)$$

$$e^{-\Gamma t}\mathcal{M}_0(\mathbf{x}) = e^{-\Gamma(\mathbf{x})t} \mathcal{M}_0(\mathbf{x}), \quad (21)$$

where $\Phi(\mathbf{x}, t)$ is the Gaussian kernel defined as

$$\Phi(\mathbf{x}, t) := \frac{1}{(4\pi Dt)^{3/2}} e^{-\mathbf{x}^2/4Dt} \quad (22)$$

and $*$ denotes the convolution over space. Note that, as opposed to the solution (16), the solution (20) and (21) is extremely fast to compute. Equation (20) simply convolves the initial state with the Gaussian kernel, which can be computed efficiently using the Fast Fourier Transform (FFT), since the boundary conditions are periodic. Equation (21) only involves exponentiation and multiplication, and thus is even faster to compute.

The resulting signal $S^h(T)$, as in the case of the finite differences, is computed by integrating the resulting discrete solution $M^h(T)$ using the trapezoidal rule and taking the absolute value of the result.

Finite element methods

The weak form (23) of the Bloch-Torrey equation is discretised using the method of lines and linear finite elements to obtain

$$\langle \mathbf{v}^h, \mathbf{u}_t^h \rangle = -\langle \mathbf{v}^h, A^h \mathbf{u}^h \rangle, \quad \forall \mathbf{v}^h \in V^h \times V^h \quad (23)$$

where $V^h \subset V = H^1$ is the space of linear finite elements on $\Omega^h \subset \Omega$. Note that Ω^h is only a proper subset of Ω when the cylinder interior is included in the domain. Otherwise, the discretisation is non-conforming.

Writing $\mathbf{u}^h = \begin{pmatrix} u^h \\ v^h \end{pmatrix}$ with u^h, v^h the vectors of nodal values of each PDE, the resulting linear system is

$$M^h \mathbf{u}_t^h = -A^h \mathbf{u}^h \quad (24)$$

where

$$\begin{aligned} A^h &= DK^h + R^h + W^h \\ M^h &= \begin{pmatrix} m^h & 0 \\ 0 & m^h \end{pmatrix} \quad K^h = \begin{pmatrix} k^h & 0 \\ 0 & k^h \end{pmatrix} \\ R^h &= \begin{pmatrix} r^h & 0 \\ 0 & r^h \end{pmatrix} \quad W^h = \begin{pmatrix} 0 & -w^h \\ w^h & 0 \end{pmatrix} \end{aligned}$$

where m^h and k^h are the usual mass and stiffness matrices for each of the coupled PDEs, and

$$r_{ij}^h := \int_{\Omega} R \phi_i \phi_j \, dx, \quad w_{ij}^h := \int_{\Omega} \omega \phi_i \phi_j \, dx.$$

Since $R(\mathbf{x})$ is strictly positive, r^h is symmetric positive definite as

$$\begin{aligned} (\mathbf{u}^h)^T r^h \mathbf{u}^h &= \sum_{i,j} \mathbf{u}_i^h \left(\int_{\Omega} R \phi_i \phi_j \, dx \right) \mathbf{u}_j^h \\ &= \int_{\Omega} R (\sum_i \mathbf{u}_i^h \phi_i)^2 \, dx \geq 0. \end{aligned}$$

Although w^h is not positive definite and A^h is not symmetric, A^h is still positive definite in the sense that

$$(\mathbf{u}^h)^T A^h \mathbf{u}^h = D(\mathbf{u}^h)^T K^h \mathbf{u}^h + (\mathbf{u}^h)^T R^h \mathbf{u}^h \geq 0$$

by the positive definiteness of K^h and R^h . This is the analogous property to the coercivity of $\langle \mathbf{u}, A\mathbf{u} \rangle$ in the weak form (13).

Time stepping for finite element methods

Asymmetric positive definite matrices do not in general have real eigenvalues. However the real part of their eigenvalues are all strictly positive. Therefore, time stepping methods for the system (24) should be chosen to be at least strongly A -stable so that iterates of the numerical solution decay at an exponential rate. The two time stepping methods which will be implemented are the L -stable backward Euler scheme and the A -stable TRBDF-2 method with parameter $\alpha = 2 - \sqrt{2}$. Additionally, the backward Euler scheme was chosen for its simplicity of implementation, and the TRBDF-2 method was chosen for its second order accuracy in time.

Backward Euler method: discretising the system (24) in time with the backward Euler method results in the time stepping iteration

$$(M^h + \delta t A^h) \mathbf{u}_+ = M^h \mathbf{u}_0 \quad (25)$$

TRBDF-2 method: discretising the system (24) in time with the TRBDF-2 method with parameter $\alpha = 2 - \sqrt{2}$ results in the time stepping scheme

$$(M^h + c_0 \delta t A^h) \mathbf{u}_\alpha = M^h \mathbf{u}_0 - c_0 \delta t A^h \mathbf{u}_0 \quad (26)$$

$$(M^h + c_0 \delta t A^h) \mathbf{u}_+ = c_+ M^h \mathbf{u}_\alpha + c_- M^h \mathbf{u}_0 \quad (27)$$

where $c_0 = 1 - 1/\sqrt{2}$, $c_{\pm} = (1 \pm \sqrt{2})/2$.

For both time stepping methods, the resulting signal $S^h(T)$ is computed by integrating the final discrete x - and y -solution components $u^h(T)$ and $v^h(T)$ and taking the magnitude of the resulting vector $S^h(T) = \|[S_x^h(T), S_y^h(T)]^T\|$.

Adaptive finite elements for the backward Euler method

For the backward Euler method, I will employ a simple *a-posteriori* error estimate for the y -component of the signal $S_y(T)$ in order to create error indicators $\eta_{\mathcal{T}}$ on each tetrahedron \mathcal{T} for use in adaptive finite elements. This simple estimate will consider only errors due to the time stepping scheme, and not errors due to the discretisation itself. The y -component is used as opposed to the full signal due to the fact that $S(T)$ is a non-linear functional of the solution, whereas $S_y(T)$ is a linear functional. Moreover, since we take $(u_0, v_0) = (0, 1)$, in practice the magnitude of $S_y(T)$ remains much larger than that of $S_x(T)$ for the simulation time T , and so the error in $S_y(T)$ is a good estimate for the error in $S(T)$.

Consider the linear system of ODEs

$$\begin{cases} u_t = Bu & \text{for } 0 < t \leq T \\ u(0) = u_0 \end{cases} \quad (28)$$

and the corresponding time-reversed dual problem

$$\begin{cases} -\phi_t = B^T \phi & \text{for } 0 \leq t < T \\ \phi(T) = \phi_0, \end{cases} \quad (29)$$

and suppose we are interested in bounding the error of a quantity of interest Q which is a linear functional of the solution at time T , taking the form

$$Q(T) = \psi \cdot u(T)$$

where ψ is a known constant vector.

Then, let equation (28) be solved using the backward Euler method, with unequal time steps allowed, and discretise u in time using the dG(0) discontinuous Galerkin method of order zero, which results in the discrete solution u^h being left-continuous piecewise constant on each time interval $(t_{n-1}, t_n]$.

Defining $e(T) = Q(T) - Q^h(T)$, it is shown in Section 4 of [14] that if ϕ_0 is set to equal ψ then

$$|e(T) \cdot \psi| \leq E_u \int_0^T \|\phi_t\| dt \leq E_u \sqrt{T \int_0^T \|\phi_t\|^2 dt} \quad (30)$$

where the second inequality follows from the Cauchy-Schwarz inequality and $E_u := \max_n \|u^h(t_n) - u^h(t_{n-1})\|$. Note also that $\phi_t = -B^T \phi$ may be used in computing the time integral of $\|\phi_t\|$.

The integral over $\|\phi_t\|^2$ may be split into a sum of integrals over the components $(\phi_t)_i^2$; define the root mean square

of the derivative components $q_i := \sqrt{\frac{1}{T} \int_0^T (\phi_t)_i^2 dt}$. Then, identifying the nodal values q_i with an appropriately defined interpolated function $q(\mathbf{x})$, the error indicator $\eta_{\mathcal{T}}$ is chosen to be

$$\eta_{\mathcal{T}} = \frac{1}{|\mathcal{T}|} \int_{\mathcal{T}} E_u T q(\mathbf{x}) d\mathbf{x}. \quad (31)$$

In order to use this estimate for the weak form of the Bloch-Torrey equation (24) with the goal of estimating the error in $Q(T) = S_y(T)$, $\psi = \mathbf{w}_{v^h}$ is set to be the vector of quadrature weights for the numerical integration of the y -component v^h of \mathbf{u}^h , and B is set to $-(M^h)^{-1}A^h$.

Then, the change of variables $\phi = M^h \Phi$ results in the time-reversed dual problem

$$\begin{cases} -M^h \Phi_t = -(A^h)^T \Phi & \text{for } 0 \leq t < T \\ \Phi(T) = (M^h)^{-1} \mathbf{w}_{v^h} \end{cases} \quad (32)$$

which may be solved using any time stepping method. Note additionally that $\phi_t = M^h \Phi_t = (A^h)^T \Phi$ is still inexpensive to compute.

Numerical experiments

Gold standard solution

As the Bloch-Torrey equation has no closed form solution, even for this simple geometry, the solution signal $S(T)$ which will be used for comparison will simply be the finite element solution computed from the system (24) on a fine mesh with small time steps **TODO: how fine and how small?**. In fact, two different gold standard signals $S(T)$ will be computed: first, with the cylinder interior excluded from the domain and Neumann boundary conditions imposed on the cylinder wall, $S_{\text{HOLLOW}}(T)$; second, with the interior of the cylinder included in the domain and no boundary conditions imposed on the cylinder wall, $S_{\text{UNION}}(T)$.

Error bound efficiency

TODO: make plot of error bound vs. true error and discuss

Results and discussion

We are now in a position to solve the Bloch-Torrey equation efficiently. We begin by discretizing the $3 \times 3 \times 3 \text{ mm}^3$ voxel into N^3 subvoxels. The subvoxels need to be small enough that diffusion effects can be captured during each time step, but large enough that solving the system is still computationally feasible. The root mean square (RMS) distance travelled by a freely diffusing particle in n -dimensions

is [15]

$$d_{rms} = \sqrt{2nD\delta t}. \quad (33)$$

If we choose $\delta t = TE/30 = 2$ ms and the diffusion constant $D = 3.037 \mu\text{m}^2/\text{ms}$ [16], the mean free diffusion distance in $n = 3$ dimensions is $d = 6.04 \mu\text{m}$. Choosing $N = 512$ gives a subvoxel size of $5.86 \mu\text{m}$, capturing diffusion effects while being computationally feasible.

Algorithm 1 details how to compute the signal at time TE using the approximate evolution equation (18).

Boundary conditions

Intuitively, periodic vs. Neumann boundary conditions on the boundary of the cubic Ω should have little effect on the resulting signal $S(T)$. Firstly, because the RMS diffusion distance (33) is much smaller than the size of the domain, and therefore only a small fraction of the signal $S(T)$ should be affected. Secondly, diffusing particles being reflected off of the boundary (the Neumann boundary condition) could similarly be interpreted as a particle from the other side of the domain diffusing into the region (periodic boundary conditions). So, as long $R(\mathbf{x})$ and $\omega(\mathbf{x})$ do not change significantly from one boundary edge to the other, the choice of boundary condition should be minimal. This intuitive result is seen in figure **TODO: figure ref + explanation**

This argument does not apply to the cylinder boundary, where $R(\mathbf{x})$ and $\omega(\mathbf{x})$ vary discontinuously. In this case, unless the vessel is sufficiently small so as to have minimal contribution to the total signal, one should expect that the Neumann boundary conditions on the cylinder wall in the FEM case vs. no boundary conditions applied in the finite differences case should have a non-negligible effect.

Conclusion

Conclusion

Acknowledgements

Acknowledgements

References

- [1] H. C. Torrey, "Bloch Equations with Diffusion Terms," *Phys. Rev.*, vol. 104, pp. 563–565, Nov. 1956.
- [2] Y.-C. N. Cheng, J. Neelavalli, and E. M. Haacke, "Limitations of Calculating Field Distributions and Magnetic Susceptibilities in MRI using a Fourier Based Method," *Phys Med Biol*, vol. 54, pp. 1169–1189, Mar. 2009.
- [3] C. Kuttler, "Reaction-diffusion equations with applications," in *Internet Seminar*, 2011.
- [4] W. Liu, *Elementary Feedback Stabilization of the Linear Reaction-Convection-Diffusion Equation and the Wave Equation*. Springer Science & Business Media, Dec. 2009. Google-Books-ID: YobRZn1HLhYC.
- [5] D. V. Nguyen, J.-R. Li, D. Grebenkov, and D. Le Bihan, "A finite elements method to solve the Bloch–Torrey equation applied to diffusion magnetic resonance imaging," *Journal of Computational Physics*, vol. 263, pp. 283–302, Apr. 2014.
- [6] C. Moler and C. Van Loan, "Nineteen Dubious Ways to Compute the Exponential of a Matrix," *SIAM Rev.*, vol. 20, pp. 801–836, Oct. 1978.
- [7] M. Caliari, P. Kandol, A. Ostermann, and S. Rainer, *Comparison of various methods for computing the action of the matrix exponential*. submitted.
- [8] C. Moler and C. Van Loan, "Nineteen Dubious Ways to Compute the Exponential of a Matrix, Twenty-Five Years Later," *SIAM Rev.*, vol. 45, pp. 3–49, Jan. 2003.
- [9] A. Al-Mohy and N. Higham, "Computing the Action of the Matrix Exponential, with an Application to Exponential Integrators," *SIAM J. Sci. Comput.*, vol. 33, pp. 488–511, Jan. 2011.
- [10] S. Gerschgorin, "über die abgrenzung der eigenwerte einer matrix," *Bulletin de l'Académie des Sciences de l'URSS. Classe des sciences mathématiques et na*, vol. 6, pp. 749–754, 1931.
- [11] G. Strang, "On the Construction and Comparison of Difference Schemes," *SIAM J. Numer. Anal.*, vol. 5, pp. 506–517, Sept. 1968.
- [12] S. MacNamara and G. Strang, "Operator Splitting," in *Splitting Methods in Communication, Imaging, Science, and Engineering*, Scientific Computation, pp. 95–114, Springer, Cham, 2016.
- [13] R. Guardiola, "Monte carlo methods in quantum many-body theories," in *Microscopic Quantum Many-Body Theories and Their Applications*, Lecture Notes in Physics, pp. 269–336, Springer, Berlin, Heidelberg, 1998. DOI: 10.1007/BFb0104529.
- [14] Eriksson K., Johnson C., and Logg A., "Adaptive Computational Methods for Parabolic Problems," *Encyclopedia of Computational Mechanics*, Nov. 2004.
- [15] T. S. Ursell, "The Diffusion Equation A Multi-dimensional Tutorial," *California Institute of Technology, Pasadena, Tech. Rep*, 2007.
- [16] M. Holz, S. R. Heil, and A. Sacco, "Temperature-dependent self-diffusion coefficients of water and six selected molecular liquids for calibration in accurate 1h NMR PFG measurements," *Phys. Chem. Chem. Phys.*, vol. 2, pp. 4740–4742, Jan. 2000.

Figures and Tables

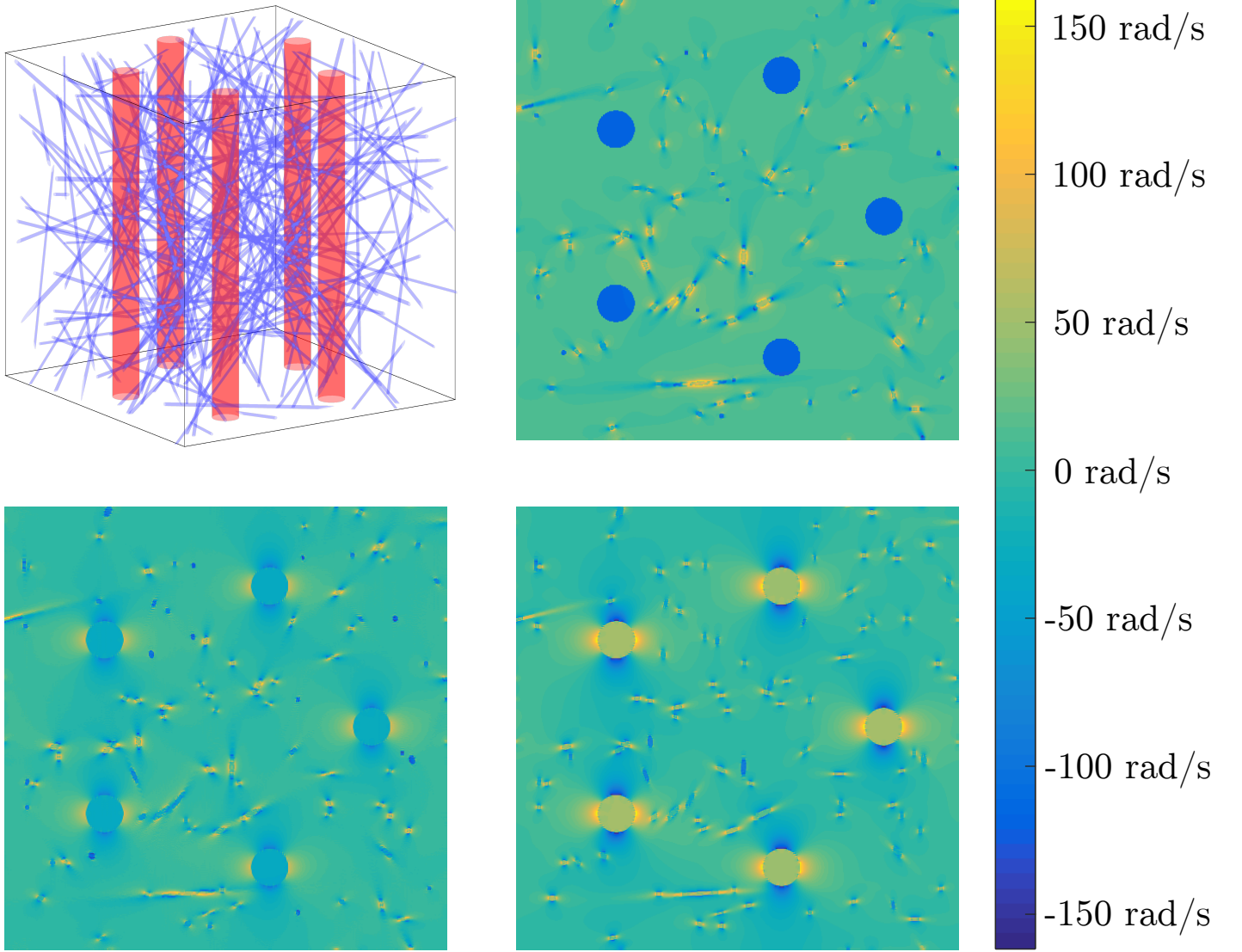


Figure 1: The top-left figure shows an example voxel geometry. The $3 \times 3 \times 3 \text{ mm}^3$ voxel is populated with an isotropic vascular bed and $L = 5$ anisotropic large vessels in the z -direction. The total volume occupied by the blood vessels is determined by the blood volume fraction BVF. The relative fraction of blood contained in the isotropic vascular bed is determined by the isotropic relative blood volume fraction iRBVF, and the amount of blood contained in the anisotropic vessels is then $\text{aRBVF} = 1 - \text{iRBVF}$. The magnetic field generated by this configuration is computed by the convolution of the susceptibility map with the unit dipole kernel. Example cross-sections of the frequency shift map $\delta\omega$ are shown for $\alpha = 0^\circ$ (top right), 45° (bottom left), and 90° (bottom right). It can be easily observed that near large vessels, the resonance frequency (i.e. the magnetic field) remains locally relatively constant compared to the resonance frequency near small vessels, which changes rapidly over short distances. Note also the increase in strength and range of inhomogeneities around the large anisotropic vessels as α increases, introducing the dependence on the angle α into the simulations.

Magnetization Propagation Algorithm

```
1: Initialize:  $\mathcal{M}_0 := i, \delta t := \text{TE}/30, k := 0$ 
2: while  $k\delta t < \text{TE}$  do
3:    $\mathcal{M}_{k+\frac{1}{2}} := e^{-\Gamma(\mathbf{x})\delta t} \mathcal{M}_k$ 
4:    $\mathcal{M}_{k+1} := \Phi(\mathbf{x}, \delta t) * \mathcal{M}_{k+\frac{1}{2}}$ 
5:   if  $(k+1)\delta t = \text{TE}/2$  then
6:      $\mathcal{M}_{k+1} := \overline{\mathcal{M}}_{k+1}$ 
7:   end if
8:    $k := k + 1$ 
9: end while
10:  $S(\text{TE}) := \int \mathcal{M}_k d^3\mathbf{x}$ 
```

Algorithm 1: Magnetization propagation algorithm used to simulate the signal $S(\text{TE})$ for a given set of free parameters CA_{PEAK} , BVF , iBVF , and L . All four free parameters are encoded solely in the complex decay rate $\Gamma(\mathbf{x})$; the rest of the algorithm does not depend on them. The notation \mathcal{M}_ν is shorthand for $\mathcal{M}(\mathbf{x}, \nu\delta t)$ throughout the algorithm. If the $\mathcal{O}(\delta t^3)$ order evolution equation (19) were used instead, line 3 should be modified to decay for only a half time step $\delta t/2$, line 4 should perform the Gaussian convolution in-place, and an extra line should be added directly following the convolution which decays for another half time step $\delta t/2$.

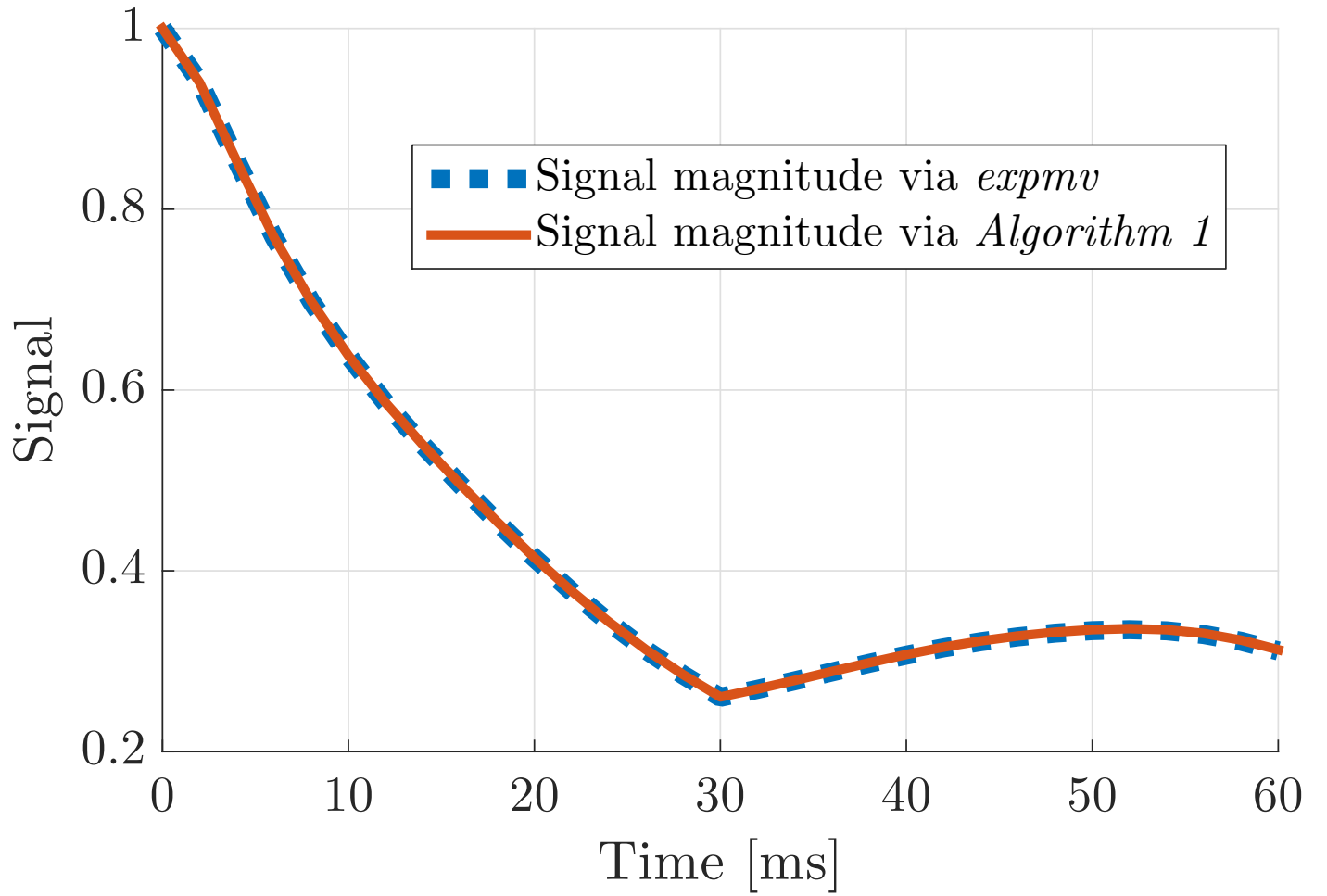


Figure 2: Comparison between solving the Bloch-Torrey equation exactly using the method of lines in conjunction with Higham's *expmv* integrator [9], and solving the Bloch-Torrey equation approximately using the two-step approximate solution as described in Algorithm 1. The signal decay through time calculation shows strong agreement between the two methods, with error values of 0.064% \pm 0.045%; the maximum error value of 0.14% occurs at 60 ms.